EPSRC Pioneering research and skills

MATHEMATICS FOR VAST DIGITAL RESOURCES

LONDON MATHEMATICAL SOCIETY 150 YEARS

NAIS Centre for Numerical Algorithms and Intelligent Software

amazon.com

Randomized Methods for Big Data: from Linear Systems to Optimization

Peter Richtárik

THE UNIVERSITY OF EDINBURGH

IEEE DSAA' 2015, Paris

Outline

1. Linear Systems
2. Minimizing Large Sums
3. An Efficient Dual Method
4. An Efficient Primal Method

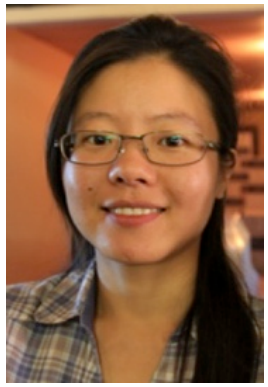
5. Distributed Optimization

Coauthors

Martin Takáč
(Lehigh)

Jakub Mareček
(IBM)

Jakub Konečný
(Edinburgh)



Zheng Qu
(University of Hong Kong)

Olivier Fercoq
(Telecom ParisTech)

Rachael Tappenden
(Johns Hopkins)

Martin Jaggi
(ETH Zurich)

Tong Zhang
(Rutgers & Baidu)

Virginia Smith
(Berkeley)

Nati Srebro
(TTI Chicago)

Jie Liu
(Lehigh)

Michael Jordan
(Berkeley)

Dominik Csba
(Edinburgh)

Robert M Gower
(Edinburgh)

1. Linear Systems

Robert Gower and P.R.
Randomized Iterative Methods for Linear Systems
arXiv:1506.03296, 2015
(to appear in SIAM Journal of Matrix Analysis and Applications)

1.1
The Problem

The Problem

$$\underbrace{\quad}_{n}$$

$$\in \mathbb{R}^n$$

$$\underbrace{\left[\begin{array}{c} \\ \\ \end{array}\right.}_{m} \quad Ax = b \quad \underbrace{\left.\begin{array}{c} \\ \\ \end{array}\right]}_{m}$$

Assumption: The system is consistent (i.e., has a solution)

We can also think of this as *m* linear equations, where the *i*th equation looks as follows:

$$\sum_{j=1}^{n} A_{ij} x_j = b_i$$

$$A_{i:} x = b_i$$

Minimizing Convex Quadratics

$$\min_{x \in \mathbb{R}^n} \left[f(x) = \frac{1}{2} \|Ax - b\|^2 \right] \quad \Rightarrow \quad \nabla f(x) = 0 \quad \Rightarrow \quad A^T A x = A^T b$$

This system is consistent

$$\min_{x \in \mathbb{R}^n} \left[f(x) = \tfrac{1}{2} x^T A x + b^T x + c \right] \quad \Rightarrow \quad \nabla f(x) = 0 \quad \Rightarrow \quad Ax = b$$

A = positive definite

This system is consistent

1.2
The Solution
(6 Ways to Skin the Cat)

TOP DEFINITION

skin the cat

Term refers to a task which has several ways by which it can be completed. Often used in the expression "there are many ways to skin the cat" or by using "skin this cat" in place of "skin the cat."

My friends and I are going to start a business, but we don't even know where to begin because there are so many ways to skin the cat.

by CRubio April 15, 2007

157 56

1. Relaxation Viewpoint
"Sketch and Project"

$$\langle x, y \rangle_B := x^T B y, \qquad \|x\|_B := \sqrt{\langle x, x \rangle_B}$$

B: Symmetric and positive definite

$$x^{t+1} = \arg\min_{x \in \mathbb{R}^n} \|x - x^t\|_B^2$$

$$\text{subject to} \quad S^T A x = S^T b$$

One Step Method: $S = m \times m$ invertible (with probability 1)

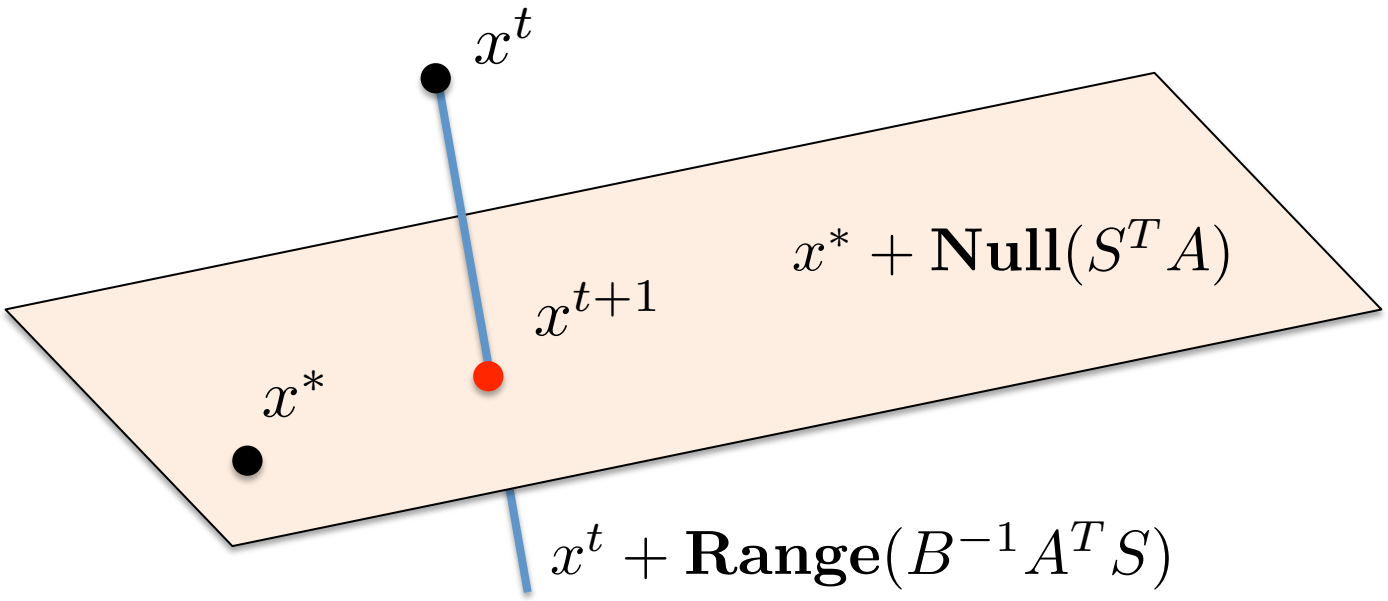
2. Optimization Viewpoint "Constrain and Approximate"

$$x^{t+1} = \arg\min_{x \in \mathbb{R}^n} \|x - x^*\|_B^2$$

$$\text{subject to} \quad x = x^t + B^{-1}A^T S y$$

$$y \quad \text{is free}$$

3. Geometric Viewpoint
"Random Intersect"



Lemma $\mathbf{Null}(S^T A)$ and $\mathbf{Range}(B^{-1}A^T S)$ are B-orthogonal complements

Proof $h \in \mathbf{Null}(S^T A) \quad \Rightarrow \quad \left\langle B^{-1}A^T S y, h \right\rangle_B = (y^T S^T A B^{-1})Bh = y^T S^T A h = 0$

$$\{x^{t+1}\} \quad = \quad \left(x^* + \mathbf{Null}(S^T A)\right) \quad \bigcap \quad \left(x^t + \mathbf{Range}(B^{-1}A^T S)\right)$$

4. Algebraic Viewpoint
"Random Linear Solve"

$$x^{t+1} = \text{solution in } x \text{ of the linear system}$$

$$S^T A x = S^T b$$

$$x = x^t + B^{-1} A^T S y$$

Unknown: *x*

Unknown: *y*

5. Algebraic Viewpoint "Random Update"

Random Update Vector

$$x^{t+1} = x^t - B^{-1}A^T S(S^T AB^{-1}A^T S)^\dagger S^T(Ax^t - b)$$

Moore-Penrose pseudo-inverse

Fact: Every (not necessarily square) real matrix M has a real pseudo-inverse $M^\dagger$.

Some properties:

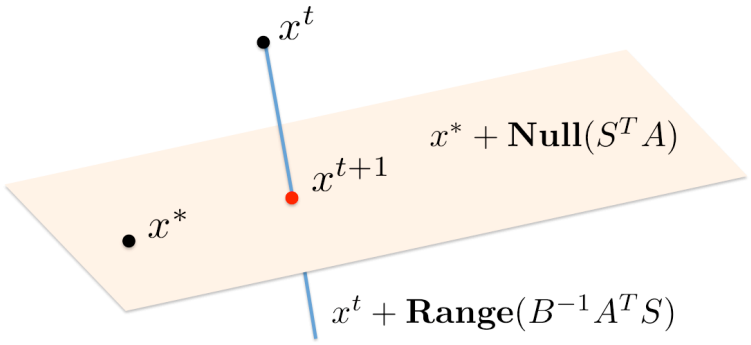
1. $MM^\dagger M = M$
2. $M^\dagger MM^\dagger = M^\dagger$
3. $(M^\top M)^\dagger M^\top = M^\dagger$
4. $(M^\top)^\dagger = (M^\dagger)^\top$
5. $(MM^\top)^\dagger = (M^\dagger)^\top M^\dagger$

6. Analytic Viewpoint
"Random Fixed Point"

$$Z := A^T S (S^T A B^{-1} A^T S)^\dagger S^T A$$

$$x^{t+1} - x^* = (I - B^{-1} Z)(x^t - x^*)$$

Random Iteration Matrix



$$(B^{-1}Z)^2 = B^{-1}Z$$
$$(I - B^{-1}Z)^2 = I - B^{-1}Z$$

$B^{-1}Z$ projects orthogonally onto $\mathbf{Range}(B^{-1}A^T S)$
$I - B^{-1}Z$ projects orthogonally onto $\mathbf{Null}(S^T A)$

Verifying that $B^{-1}Z$ is a Projection

$$M^\dagger \qquad M \qquad M^\dagger$$

$$\begin{aligned}(B^{-1}Z)^2 &= B^{-1}A^TS(S^TAB^{-1}A^TS)^\dagger S^TAB^{-1}A^TS(S^TAB^{-1}A^TS)^\dagger S^TA \\ &= B^{-1}A^TS(S^TAB^{-1}A^TS)^\dagger S^TA \\ &= B^{-1}Z\end{aligned}$$

$$Z := A^TS(S^TAB^{-1}A^TS)^\dagger S^TA$$

$$M^\dagger M M^\dagger = M^\dagger$$

Eigenvalues of $B^{-1}Z$ are in {0,1}

1.3
Complexity

Complexity / Convergence

Theorem [RG'15] For every solution x^* of $Ax = b$ we have

$$\mathbf{E}\left[x^{t+1} - x^*\right] = \left(I - B^{-1}\mathbf{E}[Z]\right)\mathbf{E}\left[x^t - x^*\right]$$

Moreover,

1. $$\left\|\mathbf{E}\left[x^t - x^*\right]\right\|_B \leq \rho^t \|x^0 - x^*\|_B$$

2. $$\mathbf{E}[Z] \succ 0$$

$$\rho := \|I - B^{-1}\mathbf{E}[Z]\|_B$$

$$\|M\|_B := \max_{\|x\|_B = 1} \|Mx\|_B$$

$$\mathbf{E}\left[\|x^t - x^*\|_B^2\right] \leq \rho^t \|x^0 - x^*\|_B^2$$

Proof of 1

$$x^{t+1} - x^* \quad = \quad (I - B^{-1}Z)(x^t - x^*)$$

Taking expectations conditioned on x^t, we get

$$\mathbf{E}[x^{t+1} - x^* \mid x^t] = (I - B^{-1}\mathbf{E}[Z])(x^t - x^*).$$

Taking expectation again gives

$$\begin{aligned} \mathbf{E}[x^{t+1} - x^*] &= \mathbf{E}\left[\mathbf{E}\left[x^{t+1} - x^* \mid x^t\right]\right] \\ &= \mathbf{E}\left[(I - B^{-1}\mathbf{E}[Z])(x^t - x^*)\right] \\ &= (I - B^{-1}\mathbf{E}[Z])\mathbf{E}[x^t - x^*]. \end{aligned}$$

Applying the norms to both sides we obtain the estimate

$$\left\|\mathbf{E}[x^{t+1} - x^*]\right\|_B \leq \underbrace{\left\|I - B^{-1}\mathbf{E}[Z]\right\|_B}_{\rho} \left\|\mathbf{E}[x^t - x^*]\right\|_B .$$

The Rate: Lower and Upper Bounds

$$d := \mathbf{Rank}(S^T A) = dim(\mathbf{Range}(B^{-1}A^T S)) = \mathbf{Tr}(B^{-1}Z)$$

Theorem [RG'15]

$$0 \leq 1 - \frac{\mathbf{E}[d]}{n} \leq \rho \leq 1$$

Insight: The method is a *contraction* (without any assumptions on *S* whatsoever). That is, things can not get worse.

Insight: The lower bound on the rate improves as the dimension of the search space in the "constrain and approximate" viewpoint grows.

Proof

$$
\begin{aligned}
\rho &= \|I - B^{-1}\mathbf{E}[Z]\|_B \\
&= \lambda_{\max}(I - B^{-1/2}\mathbf{E}[Z]B^{-1/2}) \\
&= 1 - \lambda_{\min}(B^{-1/2}\mathbf{E}[Z]B^{-1/2}) \\
&= 1 - \lambda_{\min}(\mathbf{E}[B^{-1/2}ZB^{-1/2}]) \\
&= 1 - \lambda_{\min}(\mathbf{E}[B^{-1}Z]) \\
&\geq 1 - \frac{\mathbf{Tr}\left(\mathbf{E}\left[B^{-1}Z\right]\right)}{n} \\
&= 1 - \frac{\mathbf{E}\left[\mathbf{Tr}(B^{-1}Z)\right]}{n}
\end{aligned}
$$

Direct calculation

$\|M\|_B := \max_{\|x\|_B=1} \|Mx\|_B$

XY and *YX* have the same spectrum

Upper bound

Min eigenvalue < Avg eigenvalue

The Rate: Sufficient Condition for Convergence

$$\rho = 1 - \lambda_{\min}(B^{-1}\mathbf{E}[Z])$$

Lemma

If $\mathbf{E}[Z]$ is invertible, then

(i) $\rho < 1$,

(ii) A has full column rank, and

(iii) x^* is unique

1.4
Special Case: Randomized Kaczmarz Method

Randomized Kaczmarz (RK) Method

M. S. Kaczmarz. **Angenaherte Auflosung von Systemen linearer Gleichungen,** *Bulletin International de l'Académie Polonaise des Sciences et des Lettres. Classe des Sciences Mathématiques et Naturelles. Série A, Sciences Mathématiques* 35, pp. 355–357, 1937

Kaczmarz method (1937)

T. Strohmer and R. Vershynin. **A Randomized Kaczmarz Algorithm with Exponential Convergence**. *Journal of Fourier Analysis and Applications* 15(2), pp. 262–278, 2009

Randomized Kaczmarz method (2009)

RK arises as a special case for parameters *B, S set as follows*:

$$B = I \qquad S = e^i = (0, \ldots, 0, 1, 0, \ldots, 0) \text{ with probability } p_i$$

$$x^{t+1} = x^t - \frac{A_{i:}x^t - b_i}{\|A_{i:}\|_2^2}(A_{i:})^T$$

RK was analyzed for $p_i = \frac{\|A_{i:}\|^2}{\|A\|_F^2}$

RK: Derivation and Rate

General Method

$$x^{t+1} = x^t - B^{-1}A^T S (S^T A B^{-1} A^T S)^\dagger S^T (Ax^t - b)$$

Special Choice of Parameters

$$B = I$$

$$\mathbf{P}(S = e^i) = p_i \rightarrow S = e^i$$

$$x^{t+1} = x^t - \frac{A_{i:}x^t - b_i}{\|A_{i:}\|_2^2}(A_{i:})^T$$

Complexity Rate

$$p_i = \frac{\|A_{i:}\|^2}{\|A\|_F^2}$$

$$\mathbf{E}\left[\|x^t - x^*\|_2^2\right] \leq \left(1 - \frac{\lambda_{\min}\left(A^T A\right)}{\|A\|_F^2}\right)^t \|x^0 - x^*\|_2^2$$

RK = SGD with a "smart" stepsize

$$Ax = b \qquad \text{vs} \qquad \min_x \frac{1}{2}\|Ax - b\|^2$$

Apply RK

$$f(x) = \sum_{i=1}^{m} p_i f_i(x) = \mathbf{E}_i\left[f_i(x)\right]$$
$$f_i(x) = \frac{1}{2p_i}(A_{i:}x - b_i)^2$$

Apply SGD

$$x^{t+1} = x^t - \frac{A_{i:}x^t - b_i}{\|A_{i:}\|_2^2}(A_{i:})^T$$

$$x^{t+1} = x^t - h^t \nabla f_i(x^t)$$
$$= x^t - \frac{h^t}{p_i}(A_{i:}x^t - b_i)(A_{i:})^T$$

RK is equivalent to applying SGD with a specific (smart!) constant stepsize!

$$x^{t+1} = \arg\min_{x \in \mathbb{R}^n} \|x - x^*\|_2^2 \quad \text{s.t.} \quad x = x^t + y(A_{i:})^T, \quad y \in \mathbb{R}$$

RK: Further Reading

D. Needell. **Randomized Kaczmarz solver for noisy linear systems.** *BIT* 50 (2), pp. 395-403, 2010

D. Needell and J. Tropp. **Paved with good intentions: analyzis of a randomized block Kaczmarz method.** *Linear Algebra and its Applications* 441, pp. 199-221, 2012

D. Needell, N. Srebro and R. Ward. **Stochastic gradient descent, weighted sampling and the randomized Kaczmarz algorithm.** *Mathematical Programming*, 2015 (arXiv:1310.5715)

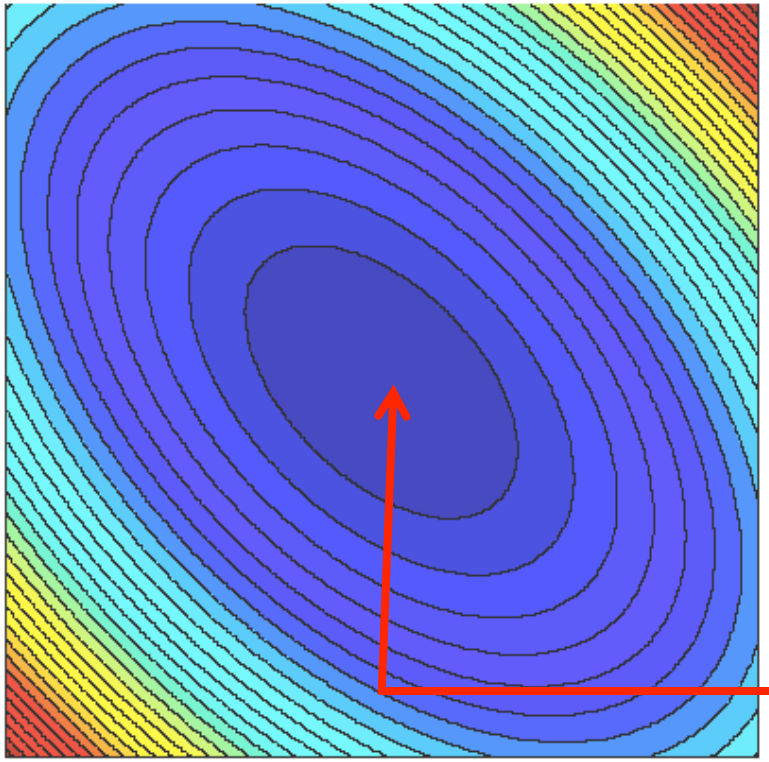
A. Ramdas. **Rows vs Columns for Linear Systems of Equations – Randomized Kaczmarz or Coordinate Descent?** *arXiv:1406.5295*, 2014

1.5
Special Case: Randomized Coordinate Descent

Coordinate Descent in 2D

Contours of a function

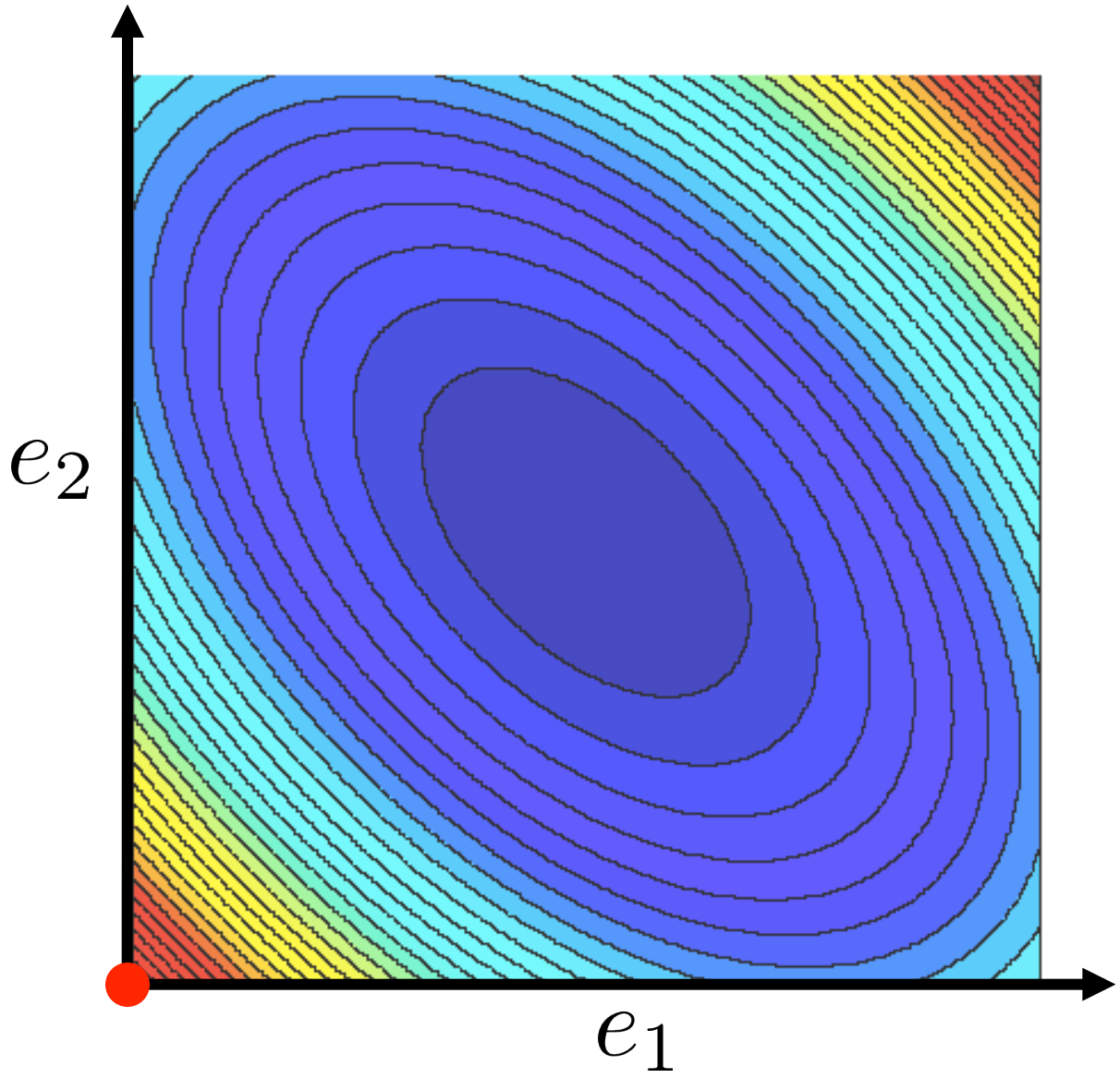
$$f : \mathbb{R}^2 \mapsto \mathbb{R}$$

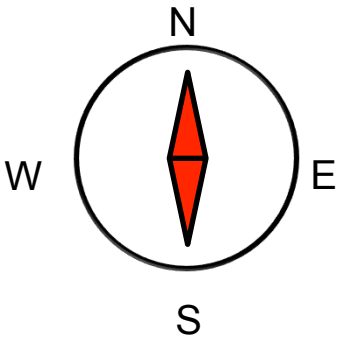


Goal:

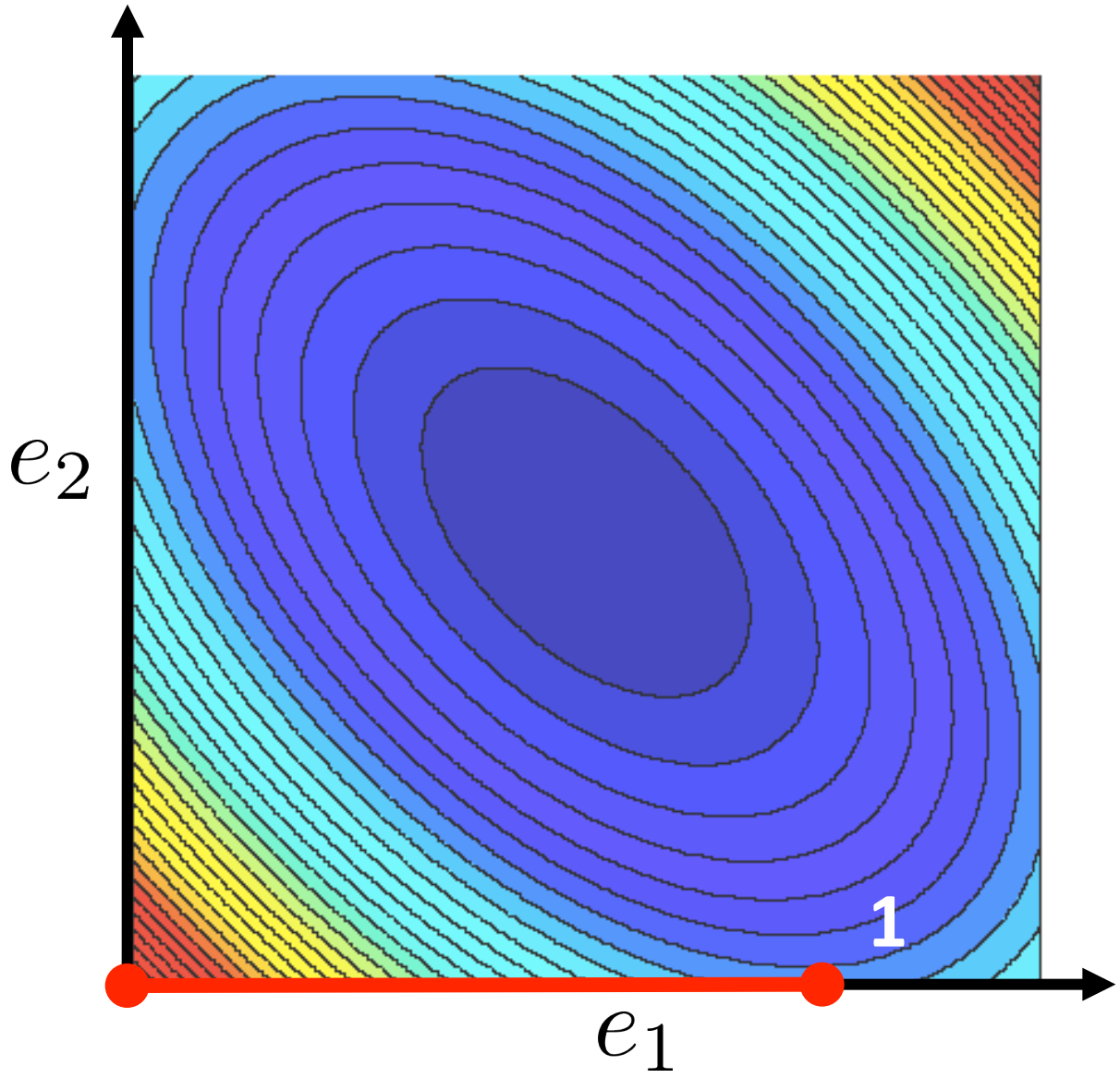
Find the minimizer

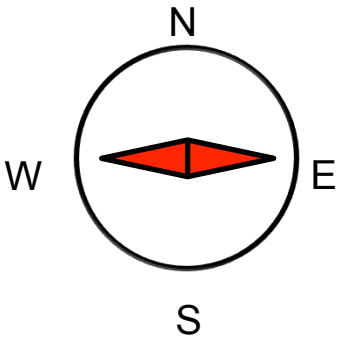
Randomized Coordinate Descent in 2D



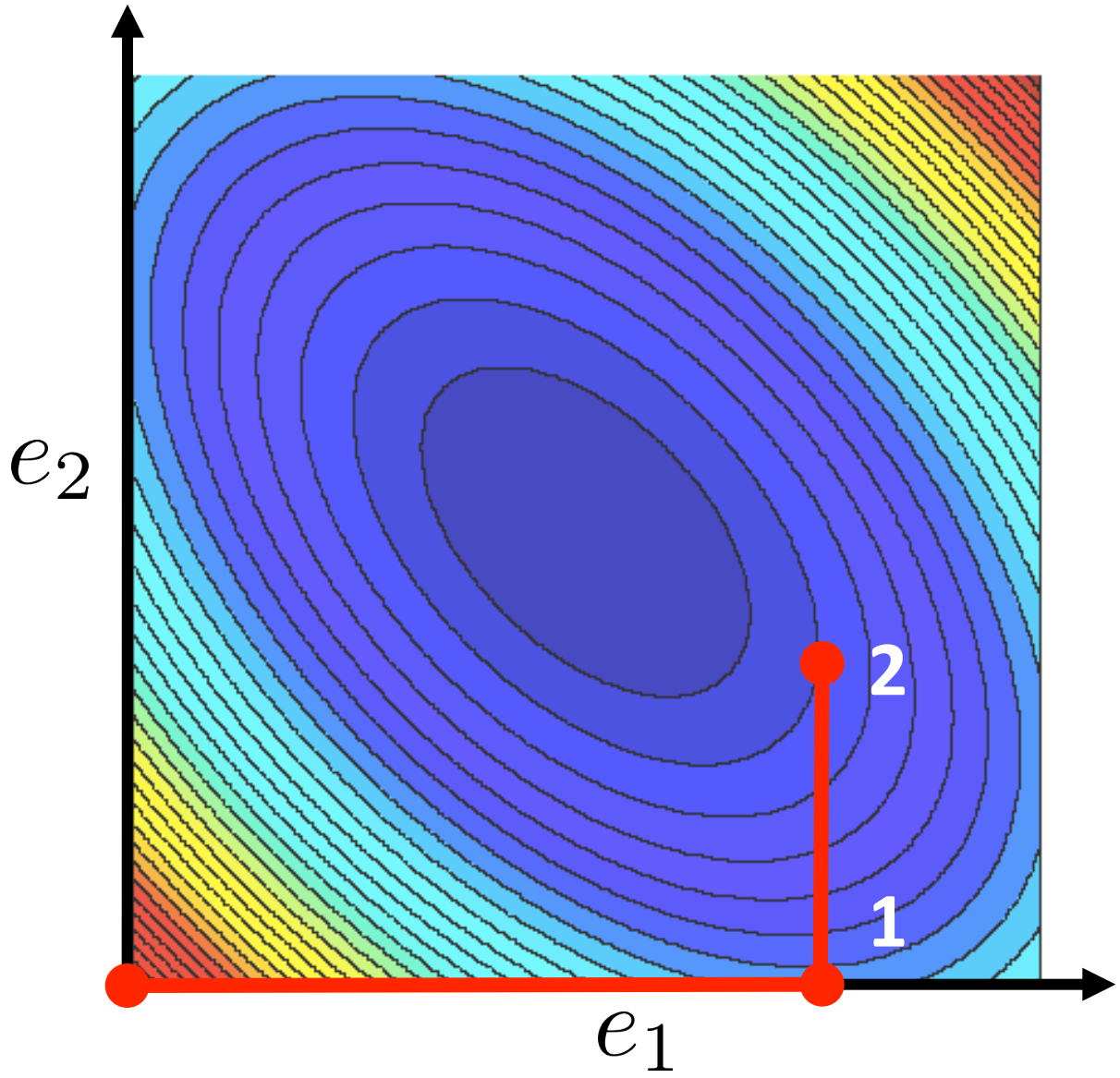


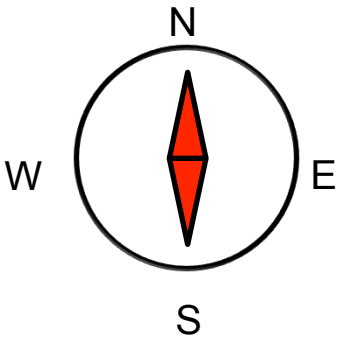
Randomized Coordinate Descent in 2D



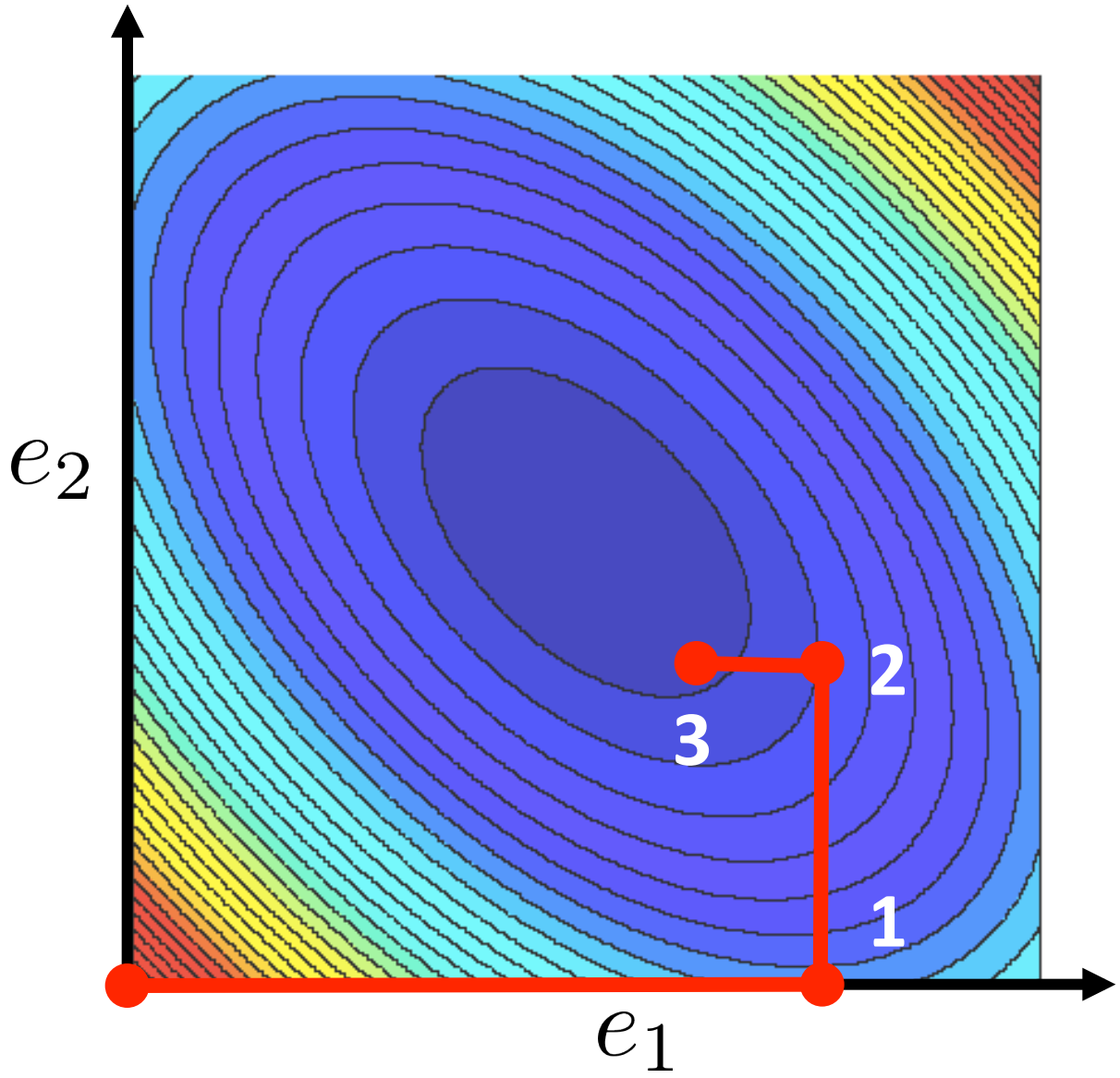


Randomized Coordinate Descent in 2D

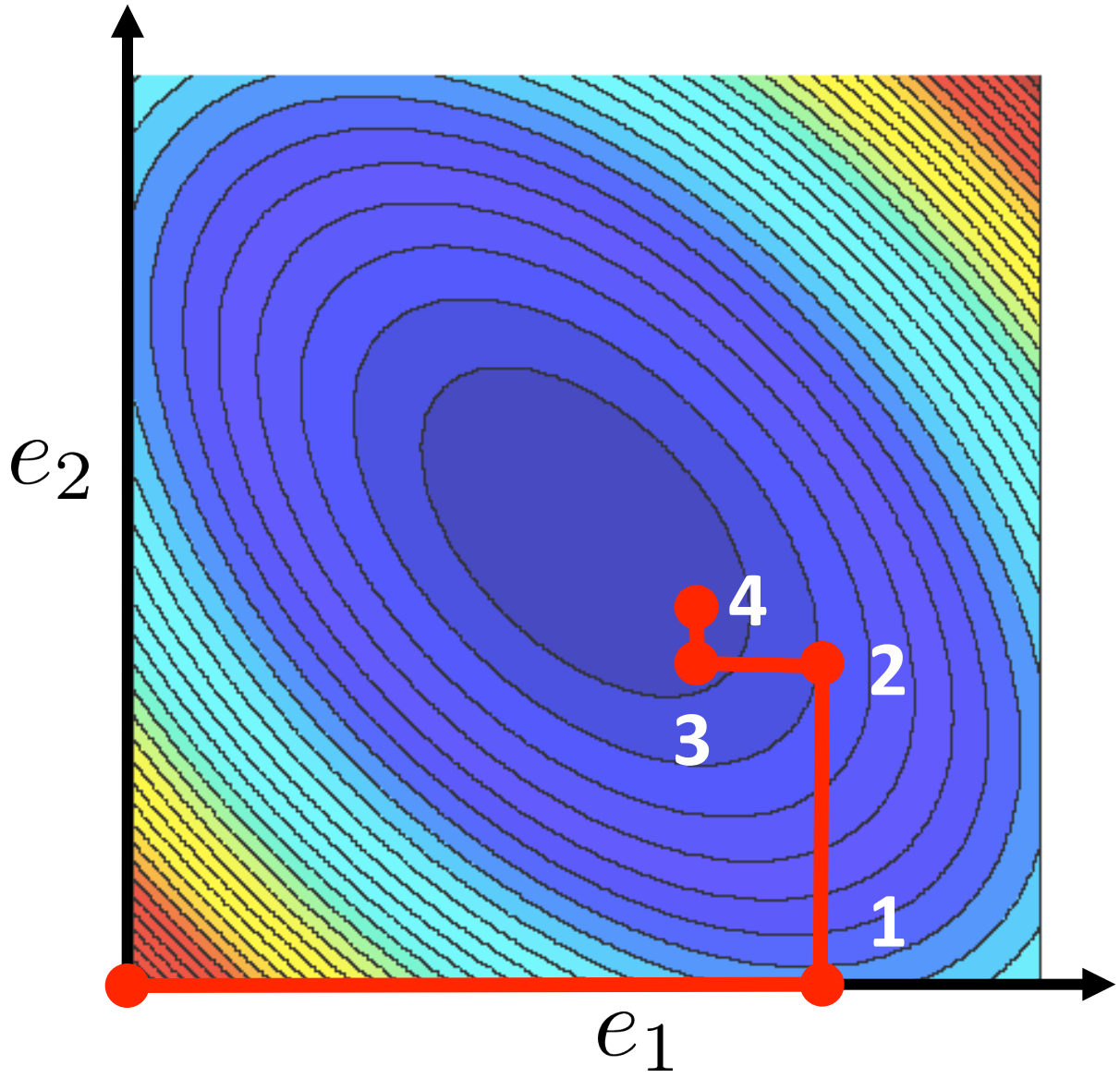




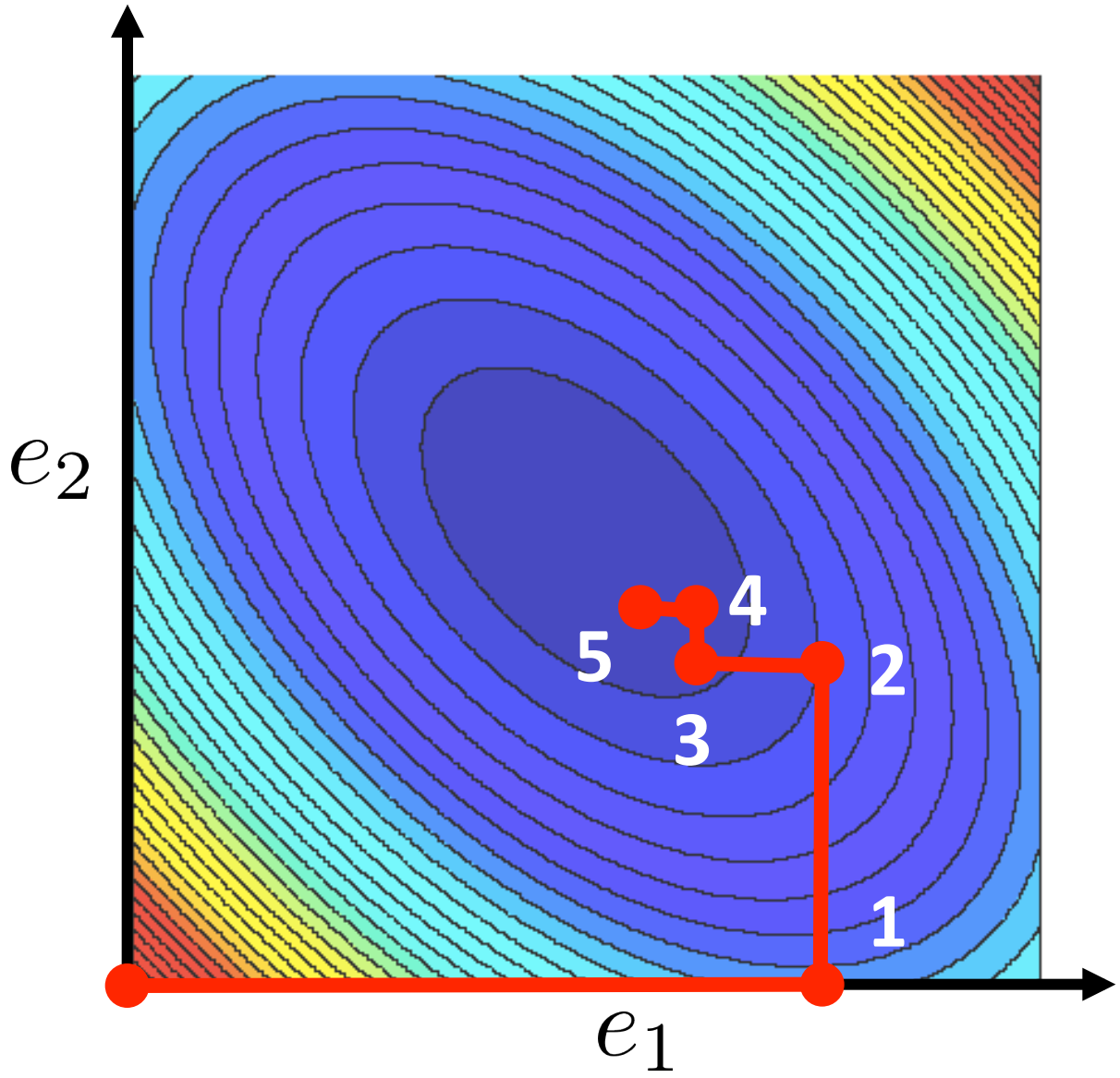
Randomized Coordinate Descent in 2D



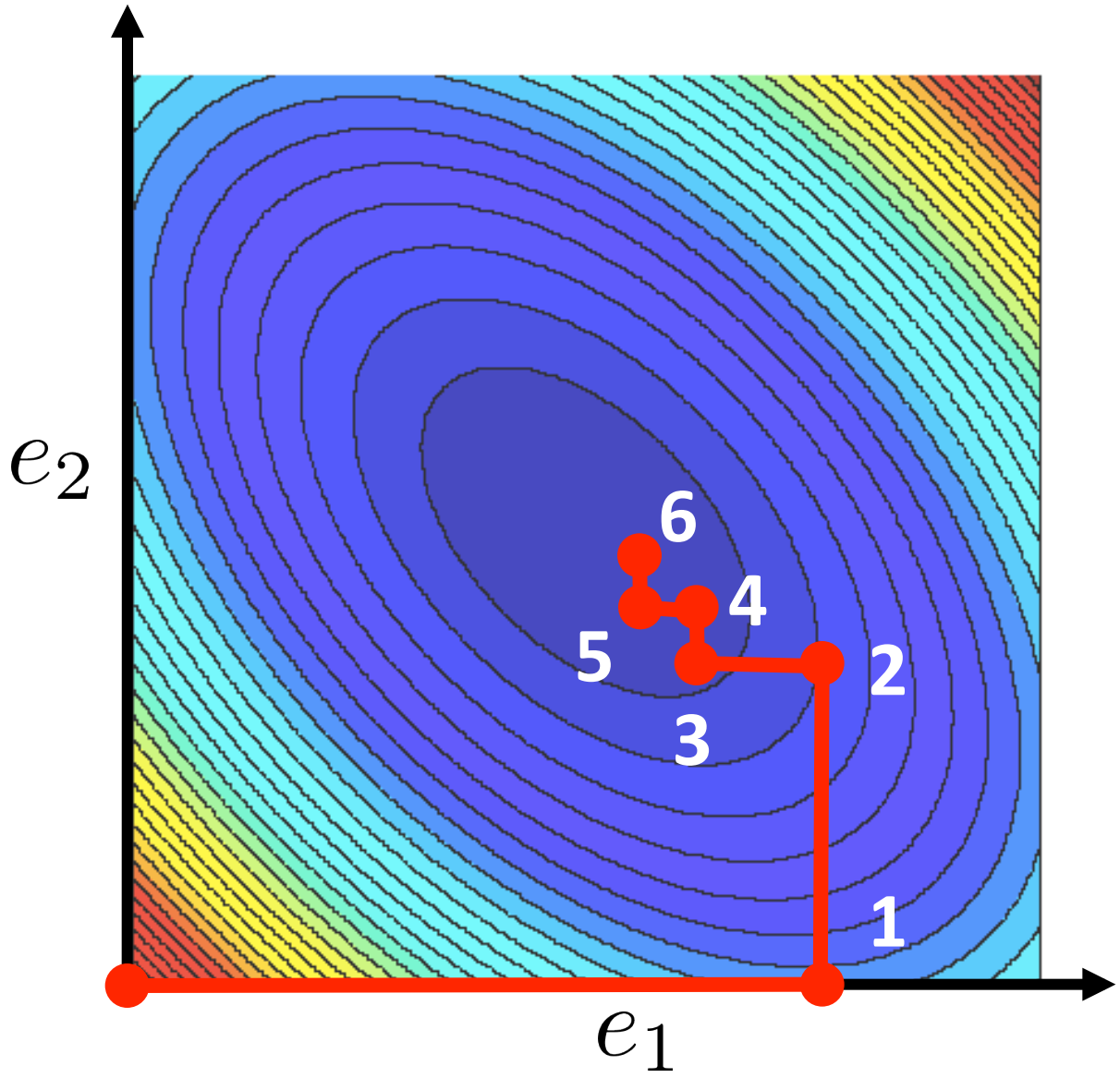
Randomized Coordinate Descent in 2D

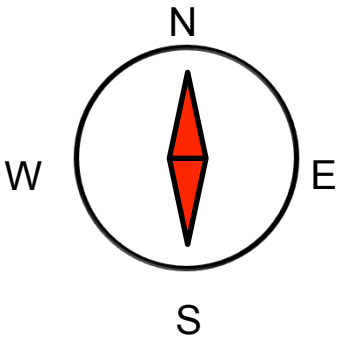


Randomized Coordinate Descent in 2D

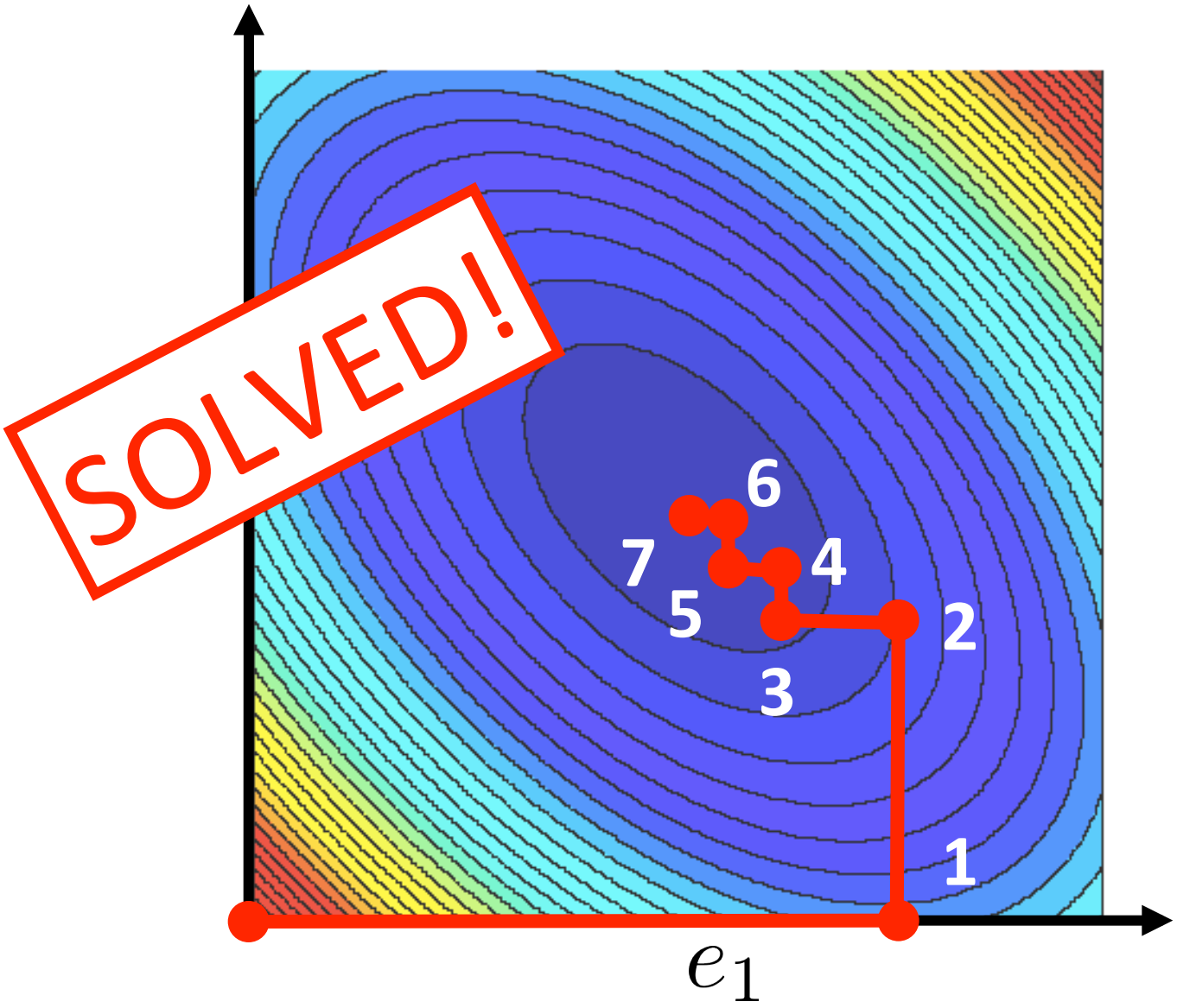


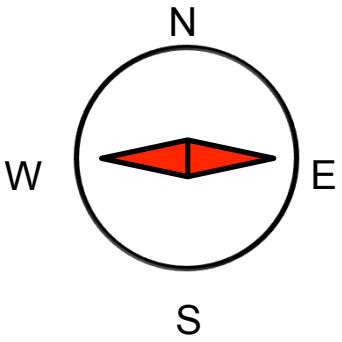
Randomized Coordinate Descent in 2D





Randomized Coordinate Descent in 2D





Randomized Coordinate Descent (RCD)

A. S. Lewis and D. Leventhal. **Randomized methods for linear constraints: convergence rates and conditioning.** *Mathematics of OR* 35(3), 641-654, 2010 (arXiv:0806.3015)

RCD (2008)

$$\min_{x \in \mathbb{R}^n} \left[f(x) = \frac{1}{2} x^T A x - b^T x \right]$$

$$x^* = A^{-1} b$$

Assume: Positive definite

RCD arises as a special case for parameters *B*, *S* *set as follows*:

$$B = A \qquad S = e^i = (0, \ldots, 0, 1, 0, \ldots, 0) \text{ with probability } p_i$$

Recall: In RK we had $B = I$

$$x^{t+1} = x^t - \frac{(A_{i:})^T x^t - b_i}{A_{ii}} e^i$$

RCD was analyzed for $p_i = \frac{A_{ii}}{\mathbf{Tr}(A)}$

RCD: Derivation and Rate

General Method

$$x^{t+1} = x^t - B^{-1}A^T S (S^T A B^{-1} A^T S)^\dagger S^T (Ax^t - b)$$

Special Choice of Parameters

$$B = A$$

$$\mathbf{P}(S = e^i) = p_i \rightarrow S = e^i$$

$$x^{t+1} = x^t - \frac{(A_{i:})^T x^t - b_i}{A_{ii}} e^i$$

Complexity Rate

$$p_i = \frac{A_{ii}}{\mathbf{Tr}(A)}$$

$$\mathbf{E}\left[\|x^t - x^*\|_A^2\right] \leq \left(1 - \frac{\lambda_{\min}(A)}{\mathbf{Tr}(A)}\right)^t \|x^0 - x^*\|_A^2$$

RCD uses "Exact Line Search"

Recall Viewpoint 2 ("Constrain and Approximate"):

$$\begin{aligned} x^{t+1} \quad = \quad & \arg\min_{x \in \mathbb{R}^n} \|x - x^*\|_B^2 \\ & \text{subject to} \quad x = x^t + B^{-1}A^T S y \\ & \qquad\qquad\quad y \quad \text{is free} \end{aligned}$$

In RCD we have:

$$B = A \quad S = e^i$$

Observation: $\|x - x^*\|_A^2 = (x - x^*)^T A(x - x^*)$

$$\begin{aligned} &= x^T Ax - 2(x^*)^T Ax + (x^*)^T Ax^* \\ &= x^T Ax - 2b^T x + b^T x^* \\ &= 2f(x) + b^T x^* \end{aligned}$$

$x^* = A^{-1}b$

$$\begin{aligned} x^{t+1} \quad = \quad & \arg\min_{x \in \mathbb{R}^n} f(x) \\ & \text{subject to} \quad x = x^t + ye^i \\ & \qquad\qquad\quad y \in \mathbb{R} \end{aligned}$$

Insight:

RCD exactly minimizes *f* along a random coordinate direction!

RCD: "Standard" Optimization Form

Yurii Nesterov. **Efficiency of coordinate descent methods on huge-scale optimization problems.** *SIAM J. on Optimization*, 22(2):341–362, 2012 (CORE Discussion Paper 2010/2)

Nesterov considered the problem:

$$\min_{x \in \mathbb{R}^n} f(x)$$

Convex and smooth

Nesterov assumed that the following inequality holds for all *x*, *h* and *i*:

$$f(x + he^i) \leq f(x) + \nabla_i f(x) h + \frac{L_i}{2} h^2$$

$$f(x) = \tfrac{1}{2} x^T A x - b^T x \quad \Rightarrow$$
$$L_i = A_{ii} \qquad \nabla_i f(x) = (A_{i:})^T x - b_i$$

Given a current iterate *x*, choosing *h* by minimizing the RHS gives:

Nesterov's RCD method:

$$x^{t+1} = x^t - \frac{1}{L_i} \nabla_i f(x^t) e^i$$

We recover RCD as we have seen it:

$$x^{t+1} = x^t - \frac{(A_{i:})^T x^t - b_i}{A_{ii}} e^i$$

1.6
Special Case: Randomized Newton Method

Randomized Newton (RN)

Z. Qu, PR, M. Takáč and O. Fercoq. **Stochastic Dual Newton Ascent for Empirical Risk Minimization.** *arXiv:1502.02268*, 2015

SDNA

$$\min_{x\in\mathbb{R}^n} \left[f(x) = \frac{1}{2}x^T Ax - b^T x \right]$$

$$x^* = A^{-1}b$$

Assume: Positive definite

RN arises as a special case for parameters *B, S set as follows*:

$$B = A \qquad S = I_{:C} \text{ with probability } p_C$$

$$p_C \geq 0 \quad \forall C \subseteq \{1,\dots,n\} \qquad \sum_{C\subseteq\{1,\dots,n\}} p_C = 1$$

RCD is special case with $p_C = 0$ whenever $|C| \neq 1$

RN: Derivation

General Method

$$x^{t+1} = x^t - B^{-1}A^T S (S^T A B^{-1} A^T S)^\dagger S^T (Ax^t - b)$$

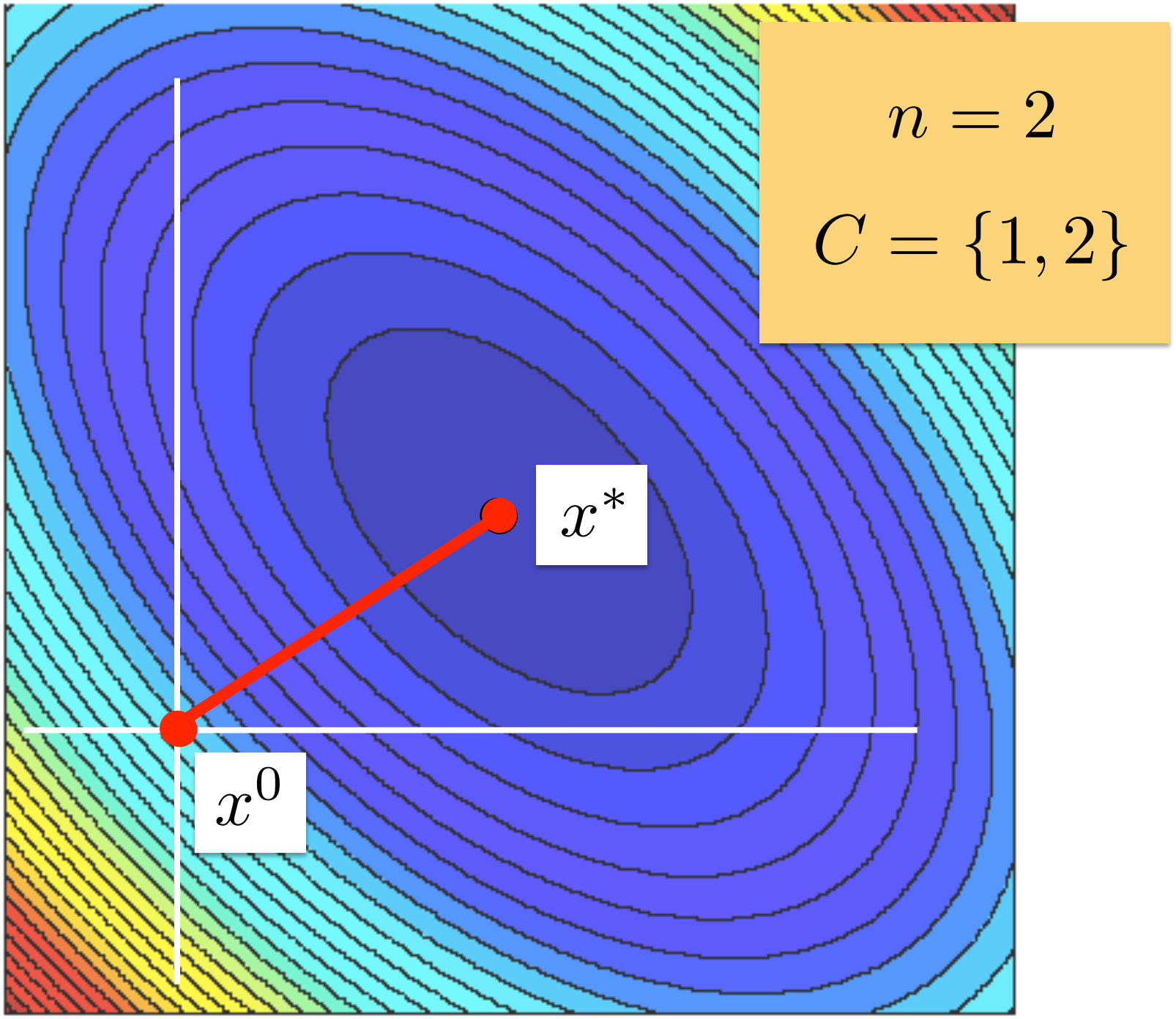
Special Choice of Parameters $B = A$

$S = I_{:C}$ with probability p_C

$$x^{t+1} = x^t - I_{:C}((I_{:C})^T A I_{:C})^{-1}(I_{:C})^T(Ax^t - b)$$

This method minimizes *f* exactly in a random subspace spanned by the coordinates belonging to *C*

Complexity Rate Will talk about this more later in the "curvature" part


$n = 2$
$C = \{1, 2\}$
x^*
x^0

1.7
Special Case:
Gaussian Descent

Gaussian Descent

General Method

$$x^{t+1} = x^t - B^{-1}A^T S (S^T A B^{-1} A^T S)^\dagger S^T (Ax^t - b)$$

Special Choice of Parameters

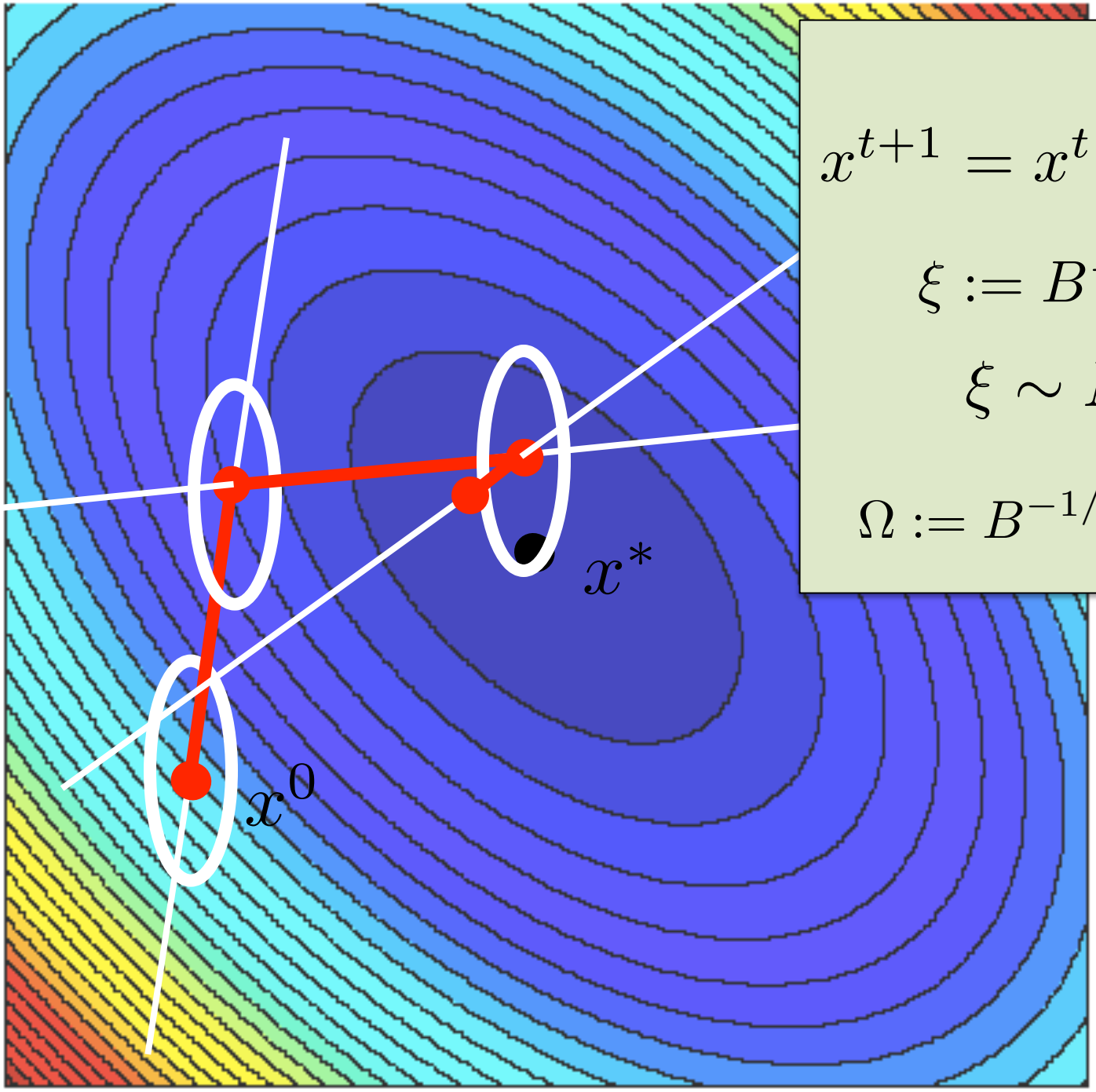
$$S \sim N(0, \Sigma)$$

Positive definite covariance matrix

$$x^{t+1} = x^t - \frac{S^T(Ax^t - b)}{S^T A B^{-1} A^T S} B^{-1} A^T S$$

Complexity Rate

$$\mathbf{E}\left[\|x^t - x^*\|_B^2\right] \leq \rho^t \|x^0 - x^*\|_B^2$$



$$x^{t+1} = x^t - h^t B^{-1/2} \xi$$

$$\xi := B^{-1/2} A^T S$$

$$\xi \sim N(0, \Omega)$$

$$\Omega := B^{-1/2} A^T \Sigma A B^{-1/2}$$

Gaussian Descent: The Rate

XY and *YX* have the same spectrum

$$
\begin{aligned}
\rho &= 1 - \lambda_{\min}(B^{-1}\mathbf{E}[Z]) \\
&= 1 - \lambda_{\min}\left(B^{-1/2}\mathbf{E}[Z]B^{-1/2}\right) \\
&= 1 - \lambda_{\min}\left(B^{-1/2}\mathbf{E}\left[A^T S(S^T AB^{-1}A^T S)^\dagger S^T A\right]B^{-1/2}\right) \\
&= 1 - \lambda_{\min}\left(\mathbf{E}\left[B^{-1/2}A^T S(S^T AB^{-1}A^T S)^\dagger S^T AB^{-1/2}\right]\right) \\
&= 1 - \lambda_{\min}\left(\mathbf{E}\left[\frac{\xi\xi^T}{\|\xi\|_2^2}\right]\right)
\end{aligned}
$$

Z

$$\xi := B^{-1/2}A^T S$$

$$\xi \sim N(0, \Omega)$$

$$\Omega := B^{-1/2}A^T \Sigma A B^{-1/2}$$

Gaussian Descent: The Rate

Lemma [GR'15]

$$\mathbf{E}\left[\frac{\xi\xi^T}{\|\xi\|_2^2}\right] \succeq \frac{2}{\pi}\frac{\Omega}{\mathbf{Tr}(\Omega)}$$

$$\rho \leq 1 - \frac{2}{\pi}\frac{\lambda_{\min}(\Omega)}{\mathbf{Tr}(\Omega)}$$

This follows from the general lower bound $1 - \frac{\mathbf{E}[d]}{n} \leq \rho$ since $d = 1$

Gaussian Descent: Further Reading

Yurii Nesterov. **Random gradient-free minimization of convex functions.** CORE Discussion Paper # 2011/1, 2011

S. U. Stitch, C. L. Muller and G. Gartner. **Optimization of convex functions with random pursuit.** SIAM Journal on Optimization 23 (2), pp. 1284-1309, 2014

S. U. Stitch. **Convex optimization with random pursuit.** PhD Thesis, ETH Zurich, 2014

1.8
Importance Sampling

Importance Sampling

Assume that S is discrete:

$$S = S_i \quad \text{with probability} \quad p_i \quad (i = 1, \dots, r)$$

Question

Consider $S_1, \dots, S_r$ fixed. How to choose the probabilities $p_1, \dots, p_r$ which optimize the convergence rate $\rho = 1 - \lambda_{\min}(B^{-1}\mathbf{E}[Z])$?

$$\max_p \left\{ \lambda_{\min}(B^{-1}\mathbf{E}[Z]) \quad \text{subject to} \quad \sum_{i=1}^r p_i = 1,\ p \geq 0 \right\}$$

- Can be reformulated as an SDP (Semidefinite Program)
- Leads to different probabilities than those proposed for RK and RCD!

$$\begin{aligned} \max_{p,t} \quad & t \\ \text{subject to} \quad & \sum_{i=1}^r p_i \left(V_i (V_i^T V_i)^\dagger V_i^T \right) \succeq t \cdot I, \\ & p \geq 0, \quad \sum_{i=1}^r p_i = 1 \end{aligned}$$

$$V_i = B^{-1/2} A^T S_i$$

RCD: Optimal Probabilities Can Lead to a Remarkable Improvement

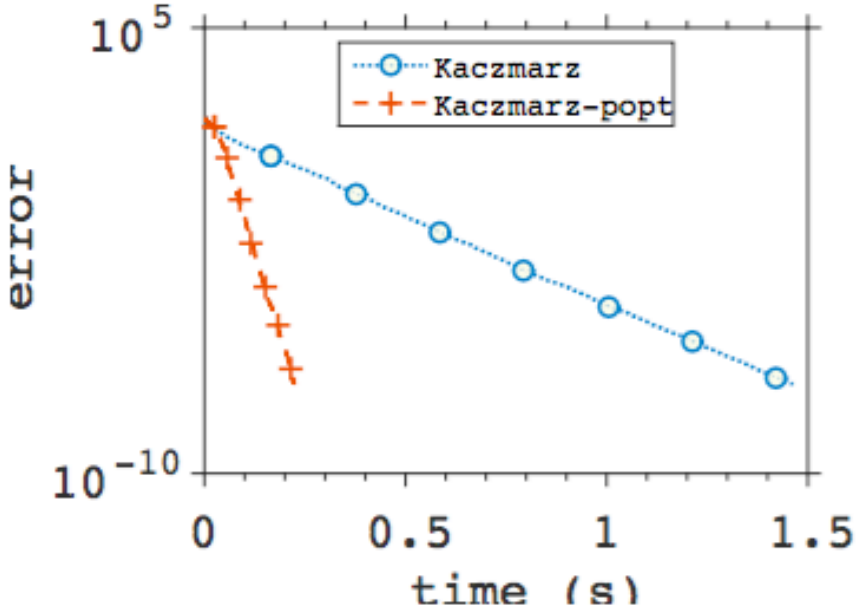
Rate for **convenient** (standard) **probabilities**

Rate for **optimal probabilities** (solving SDP)

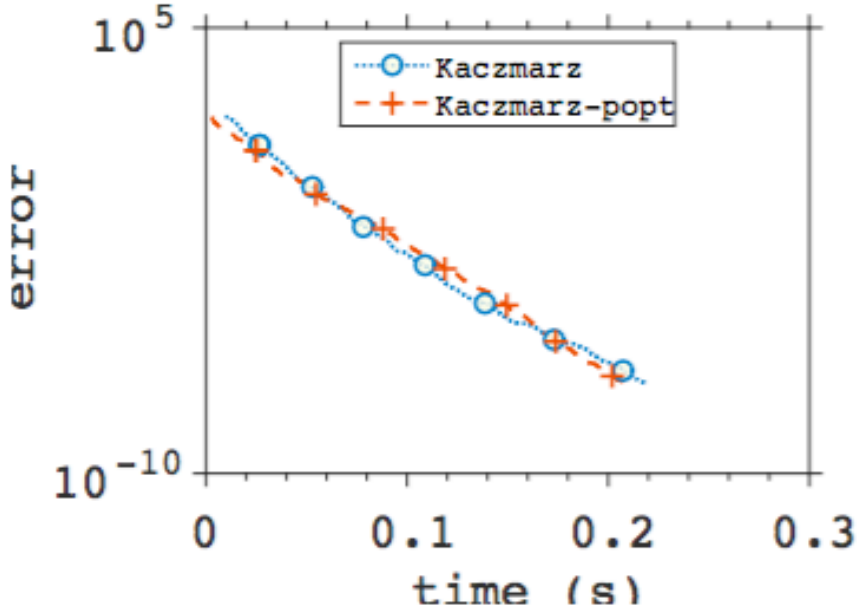
Lower bound on the rate

data set	ρ_c	ρ^*	$1-1/n$
rand(50,50)	$1-2\cdot 10^{-6}$	$1-3.05\cdot 10^{-6}$	$1-2.10^{-2}$
mushrooms-ridge	$1-5.86\cdot 10^{-6}$	$1-7.15\cdot 10^{-6}$	$1-8.93\cdot 10^{-3}$
aloi-ridge	$1-2.17\cdot 10^{-7}$	$1-1.26\cdot 10^{-4}$	$1-7.81\cdot 10^{-3}$
liver-disorders-ridge	$1-5.16\cdot 10^{-4}$	$1-8.25\cdot 10^{-3}$	$1-1.67\cdot 10^{-1}$
covtype.binary-ridge	$1-7.57\cdot 10^{-14}$	$1-1.48\cdot 10^{-6}$	$1-1.85\cdot 10^{-2}$

RK: Convenient vs Optimal

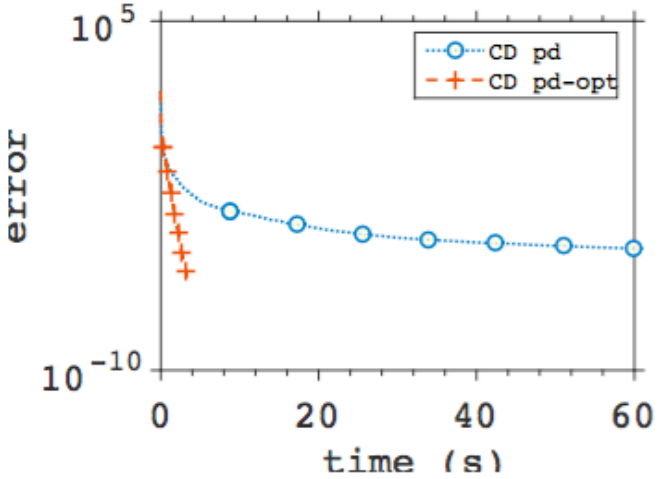


(a) liver-disorders-popt-k

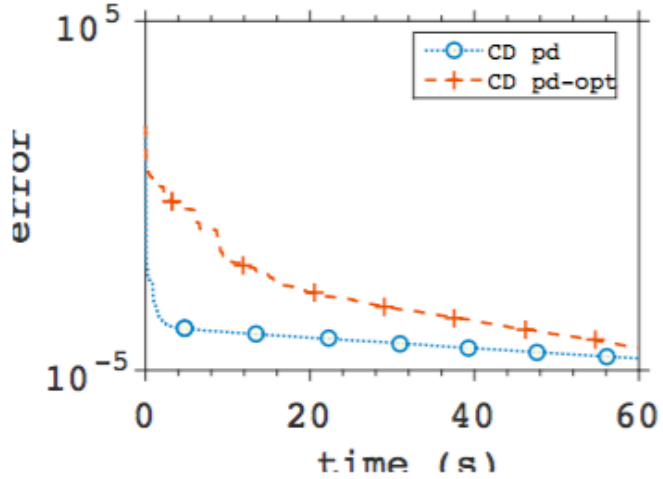


(b) rand(500,100)

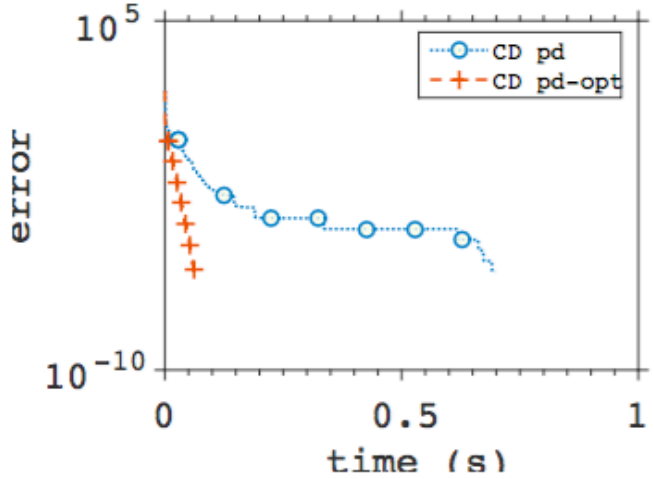
RCD: Convenient vs Optimal



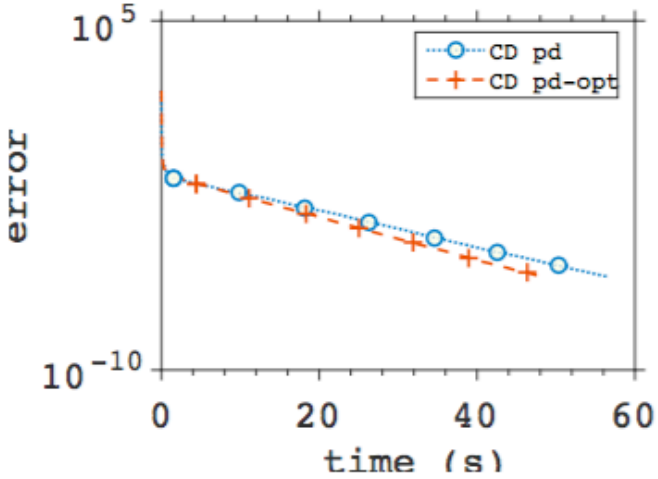
(a) aloi



(b) covtype.libsvm.binary



(c) liver-disorders-ridge

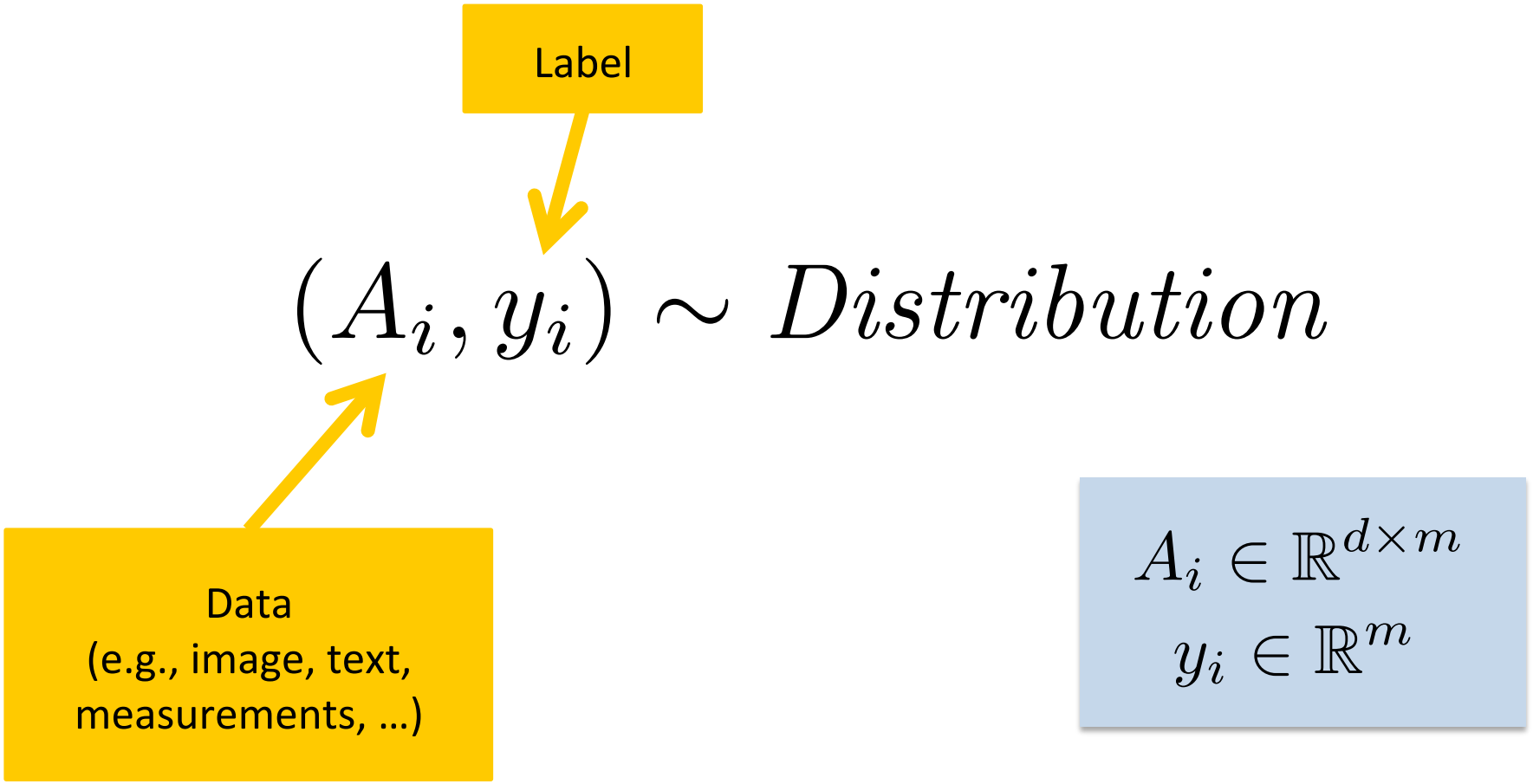


(d) mushrooms-ridge-opt

2. Minimizing Large Sums

2.1
Training Linear Predictors

Statistical Nature of Data



Prediction of Labels from Data

Find $w \in \mathbb{R}^d$ ← Linear predictor

Such that when (data, label) pair is drawn from the distribution

$$(A_i, y_i) \sim Distribution$$

Then

$$A_i^\top w \approx y_i$$

Predicted label → $A_i^\top w$ ← True label

Measure of Success

$$loss(a, b)$$

We want the expected loss (=risk) to be small:

$$\mathbf{E}\left[loss(A_i^\top w, y_i)\right]$$

$$(A_i, y_i) \sim Distribution$$

Finding a Linear Predictor via Empirical Risk Minimization (ERM)

Draw i.i.d. data (samples) from the distribution

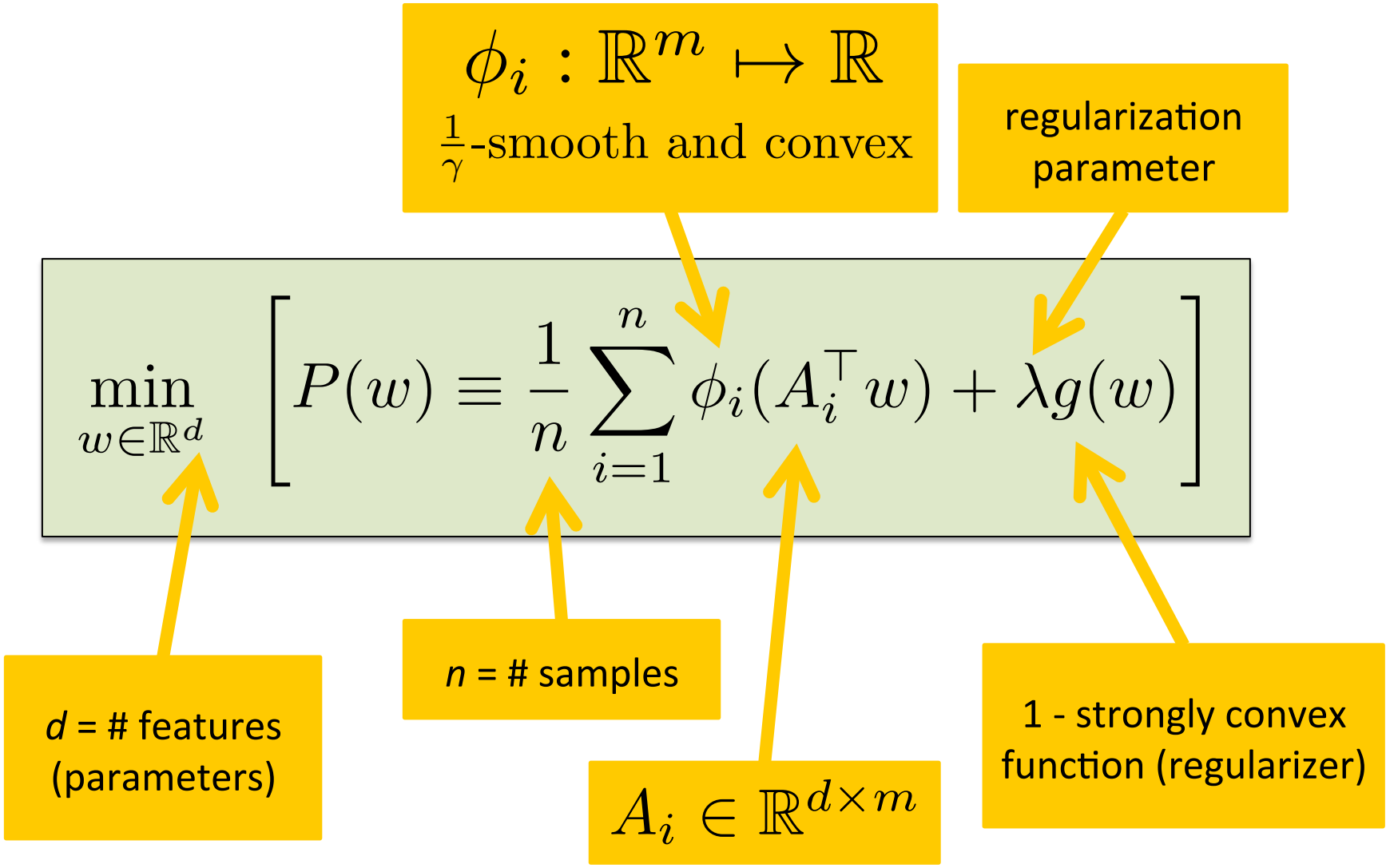
$$(A_1, y_1), (A_2, y_2), \ldots, (A_n, y_n) \sim Distribution$$

Output predictor which minimizes the empirical risk:

$$\min_{w \in \mathbb{R}^d} \frac{1}{n} \sum_{i=1}^{n} loss(A_i^\top w, y_i)$$

2.2
Primal and Dual Problems

ERM: Primal Problem



ERM: Dual Problem

$$D(\alpha) \equiv -\lambda g^* \left(\frac{1}{\lambda n} \sum_{i=1}^{n} A_i \alpha_i \right) - \frac{1}{n} \sum_{i=1}^{n} \phi_i^*(-\alpha_i)$$

$\in \mathbb{R}^m$ (over α_i)

$\in \mathbb{R}^d$ (over $\frac{1}{\lambda n} \sum_{i=1}^{n} A_i \alpha_i$)

1 – smooth & convex

γ - strongly convex

$$g^*(w') = \max_{w \in \mathbb{R}^d} \left\{ (w')^\top w - g(w) \right\}$$

$$\phi_i^*(a') = \max_{a \in \mathbb{R}^m} \left\{ (a')^\top a - \phi_i(a) \right\}$$

$$\max_{\alpha = (\alpha_1, \ldots, \alpha_n) \in \mathbb{R}^N = \mathbb{R}^{nm}} D(\alpha)$$

$\alpha_1 \in \mathbb{R}^m$, $\alpha_n \in \mathbb{R}^m$

Two Approaches

- Work with the **primal problem**
  - Process one loss function (= one example) at a time
  - Type of methods: stochastic gradient descent (modern variants: SAG, SVRG, S2GD, mS2GD, SAGA, S2CD, MISO, FINITO, …)
- Work with the **dual problem**
  - Process one dual variable (=one example) at a time
  - Type of methods: randomized coordinate descent (modern variants: RCDM, PCDM, Shotgun, SDCA, APPROX, Quartz, ALPHA, SDNA, SPDC, ASDCA, …)

3. An Efficient Dual Method

Zheng Qu, P.R. and Tong Zhang
Randomized dual coordinate ascent with arbitrary sampling
In NIPS 2015 (arXiv:1411.5873)

3.1
Two Assumptions

Assumption 1

The loss functions $\phi_i : \mathbb{R}^m \mapsto \mathbb{R}$ are $\frac{1}{\gamma}$-smooth:

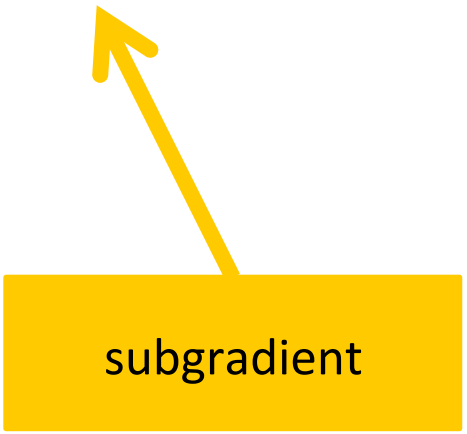
$$\|\nabla\phi_i(a) - \nabla\phi_i(a')\| \leq \frac{1}{\gamma}\|a - a'\|, \quad a, a' \in \mathbb{R}^m$$

Lipschitz constant of the gradient of the function

Assumption 2

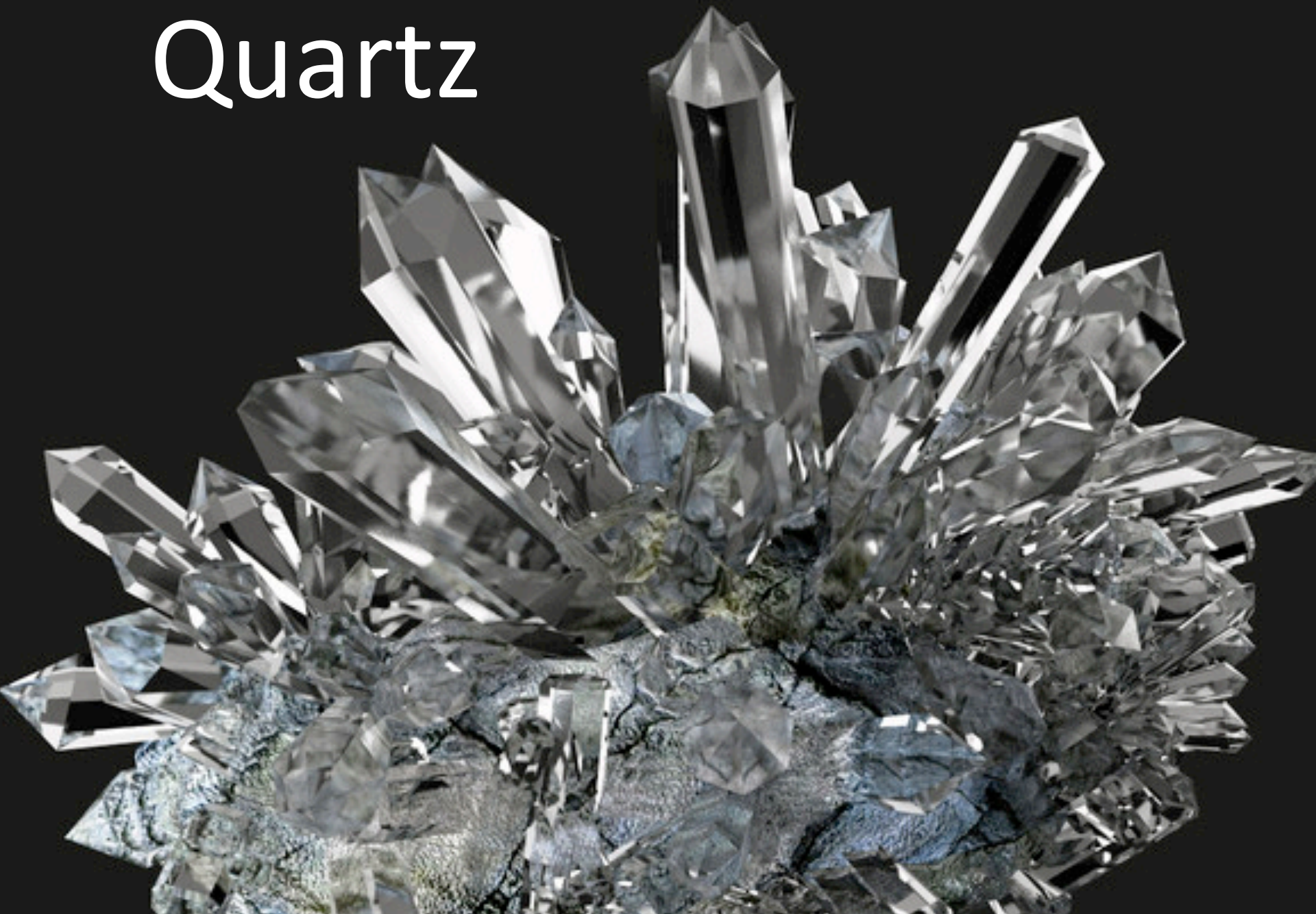
Regularizer is 1-strongly convex

$$g(w) \geq g(w') + \langle \nabla g(w'), w - w' \rangle + \frac{1}{2}\|w - w'\|^2, \quad w, w' \in \mathbb{R}^d$$



3.2
The Algorithm

Quartz



Fenchel Duality

$$\bar{\alpha} = \frac{1}{\lambda n} \sum_{i=1}^{n} A_i \alpha_i$$

$$P(w) - D(\alpha) \;=\; \lambda \left(g(w) + g^* \left(\bar{\alpha} \right) \right) + \frac{1}{n} \sum_{i=1}^{n} \phi_i(A_i^\top w) + \phi_i^*(-\alpha_i) =$$

$$\lambda (g(w) + g^* \left(\bar{\alpha} \right) - \langle w, \bar{\alpha} \rangle) + \frac{1}{n} \sum_{i=1}^{n} \phi_i(A_i^\top w) + \phi_i^*(-\alpha_i) + \left\langle A_i^\top w, \alpha_i \right\rangle$$

$$\geq 0 \quad \Leftarrow \quad \textbf{Weak duality} \quad \Rightarrow \quad \geq 0$$

Optimality conditions

$$w = \nabla g^*(\bar{\alpha}) \qquad \alpha_i = -\nabla \phi_i(A_i^\top w)$$

The Algorithm

$$(\alpha^t, w^t) \quad \Rightarrow \quad (\alpha^{t+1}, w^{t+1})$$

Quartz: Bird's Eye View

STEP 1: PRIMAL UPDATE

$$w^{t+1} \leftarrow (1-\theta)w^t + \theta \nabla g^*(\bar{\alpha}^t)$$

STEP 2: DUAL UPDATE

Choose a random set S_t of dual variables

For $i \in S_t$ do

$$p_i = \mathbf{P}(i \in S_t)$$

$$\alpha_i^{t+1} \leftarrow \left(1 - \frac{\theta}{p_i}\right)\alpha_i^t + \frac{\theta}{p_i}\left(-\nabla \phi_i(A_i^\top w^{t+1})\right)$$

3.3
Other Stochastic Dual Methods for ERM

Randomized Dual Coordinate Ascent Methods for ERM

Algorithm	1-nice	1-optimal	τ-nice	arbitrary	additional speedup	direct p-d analysis	acceleration
SDCA	●						
mSDCA	●		●		●		
ASDCA	●		●				●
AccProx-SDCA	●						●
DisDCA	●		●				
Iprox-SDCA	●	●					
APCG	●						●
SPDC	●	●	●			●	●
Quartz	●	●	●	●	●	●	

SDCA: SS Shwartz & T Zhang, 09/2012
mSDCA M Takac, A Bijral, P R & N Srebro, 03/2013
ASDCA: SS Shwartz & T Zhang, 05/2013
AccProx-SDCA: SS Shwartz & T Zhang, 10/2013
DisDCA: T Yang, 2013
Iprox-SDCA: P Zhao & T Zhang, 01/2014
APCG: Q Lin, Z Lu & L Xiao, 07/2014
SPDC: Y Zhang & L Xiao, 09/2014
Quartz: Z Qu, P R & T Zhang, 11/2014

3.4
Complexity

Assumption 3
(Expected Separable Overapproximation)

Parameters $v_1, \ldots, v_n$ satisfy:

$$\mathbf{E}\left\|\sum_{i\in S_t} A_i\alpha_i\right\|^2 \leq \sum_{i=1}^{n} p_i v_i \|\alpha_i\|^2$$

inequality must hold for all

$$\alpha_1, \ldots, \alpha_n \in \mathbb{R}^m$$

$$p_i = \mathbf{P}(i \in S_t)$$

Complexity

$$\theta = \min_i \frac{p_i \lambda \gamma n}{v_i + \lambda \gamma n}$$

Theorem [Qu, R & Zhang 14]

$$\mathbf{E}[P(w^t) - D(\alpha^t)] \leq (1-\theta)^t (P(w^0) - D(\alpha^0))$$

$$t \geq \max_i \left(\frac{1}{p_i} + \frac{v_i}{p_i \lambda \gamma n} \right) \log \left(\frac{P(w^0) - D(\alpha^0)}{\epsilon} \right)$$

$$\Rightarrow \quad \mathbf{E}\left[P(w^t) - D(\alpha^t)\right] \leq \epsilon$$

Example

Data: $n = 7 \times 10^5$

$\gamma = \frac{1}{4} \quad v_i \equiv \lambda_{\max}(A_i^\top A_i) \leq 1$

Method: $|S_t| \equiv 1 \quad p_i = \frac{1}{n} \quad \lambda = \frac{1}{n}$

$$(1-\theta)^n = 0.8187$$

$$(1-\theta)^{12n} = 0.0907 < \tfrac{1}{10}$$

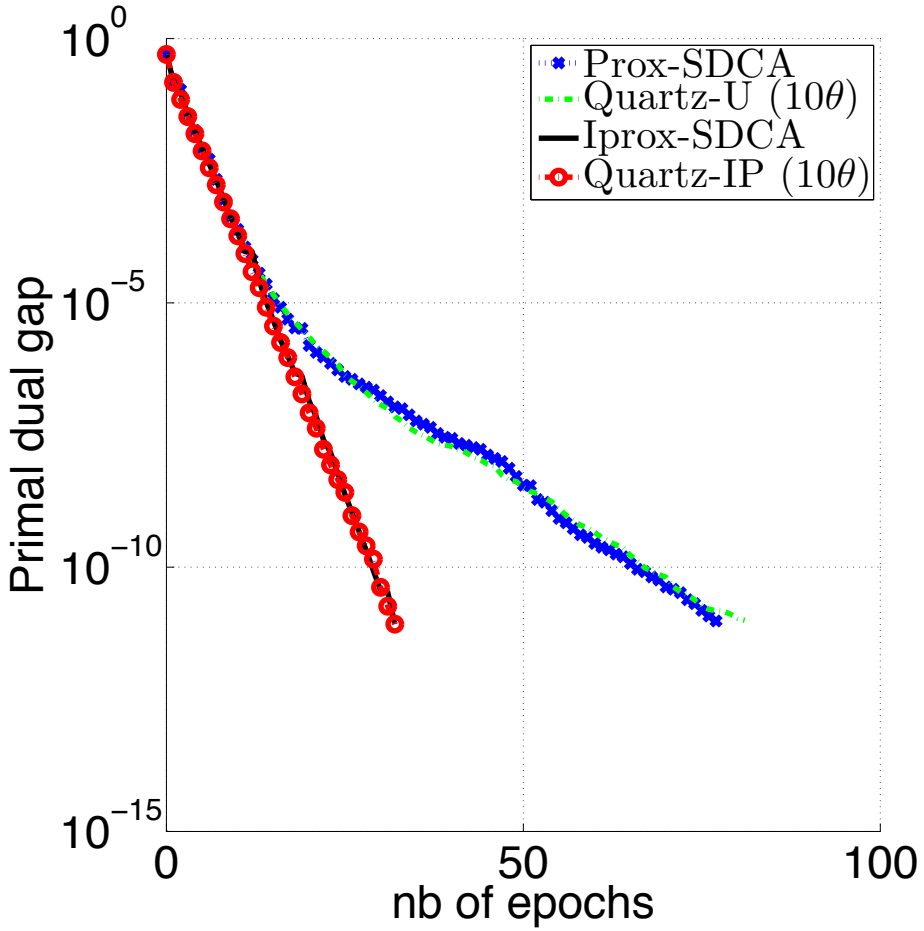
3.5
Updating One Dual Variable at a Time

Complexity of Quartz specialized to serial sampling

Optimal sampling	$n + \frac{\frac{1}{n}\sum_{i=1}^{n} L_i}{\lambda\gamma}$
Uniform sampling	$n + \frac{\max_i L_i}{\lambda\gamma}$

$$L_i \equiv \lambda_{\max}\left(A_i^\top A_i\right)$$

Experiment: Quartz vs SDCA, uniform vs optimal sampling



$$\text{Data} = \texttt{cov1}, \quad n = 522,911, \quad \lambda = 10^{-6}$$

4. An Efficient Primal Method

S. Shalev-Shwartz
SDCA without Duality, *NIPS 2015* (arXiv:1502.06177)

Dominik Csiba and P.R.
Primal method for ERM with flexible mini-batching schemes and non-convex losses, *arXiv:1506.02227,* 2015

4.1
Empirical Risk Minimization

Primal Problem: ERM

$$\phi_i : \mathbb{R}^m \mapsto \mathbb{R}$$

$\frac{1}{\gamma}$-smooth and convex

regularization parameter

$$\min_{w \in \mathbb{R}^d} \left[P(w) \equiv \frac{1}{n} \sum_{i=1}^{n} \phi_i(A_i^\top w) + \frac{\lambda}{2} \|w\|_2^2 \right]$$

d = # features (parameters)

n = # samples

$A_i \in \mathbb{R}^{d \times m}$

We had a general 1-strongly convex function *g* here before

Dual Problem

$$D(\alpha) \equiv -\lambda$$

$$g^*(w') = \max_{w \in \mathbb{R}^d} \left\{ (w')^\top w \right.$$

1 – smooth & convex

Goal: An efficient algorithm which naturally operates in the primal space (i.e., on the primal problem) only

The method will have the "same" theoretical guarantee as Quartz

$$\max_{\alpha=(\alpha_1,\ldots,\alpha_n)\in\mathbb{R}^N=\mathbb{R}^{nm}} D(\alpha)$$

$$\in \mathbb{R}^m \quad \in \mathbb{R}^m$$

4.2
The Algorithm

Motivation I

w^* is optimal

$$0 = \nabla P(w^*) = \left(\frac{1}{n} \sum_{i=1}^{n} A_i \nabla \phi_i(A_i^\top w^*) \right) + \lambda w^*$$

$$w^* = \frac{1}{\lambda n} \sum_{i=1}^{n} A_i \alpha_i^*$$

$$\alpha_i^* := -\nabla \phi_i(A_i^\top w^*)$$

Motivation II

Algorithmic Ideas:

1. Simultaneously search for both w^* and $\alpha_1^*, \ldots, \alpha_n^*$

2. Try to do "something like"

$$\alpha_i^{t+1} \leftarrow -\nabla \phi_i(A_i^\top w^t)$$

3. Maintain the relationship

$$w^t = \frac{1}{\lambda n} \sum_{i=1}^{n} A_i \alpha_i^t$$

Does not quite work: too "greedy"

The Algorithm: dfSDCA

STEP 0: INITIALIZE

Choose $\alpha_1^0, \ldots, \alpha_n^0 \in \mathbb{R}^m$

$$w^0 = \frac{1}{\lambda n} \sum_{i=1}^{n} A_i \alpha_i^0$$

Initialize the relationship

STEP 1: "DUAL" UPDATE

Choose a random set S_t of "dual variables"

For $i \in S_t$ do

$$\alpha_i^{t+1} \leftarrow \left(1 - \frac{\theta}{p_i}\right) \alpha_i^t + \frac{\theta}{p_i} \left(-\nabla \phi_i(A_i^\top w^t)\right)$$

Controlling "greed" by taking a convex combination

$$\theta = \min_i \frac{p_i \lambda \gamma n}{v_i + \lambda \gamma n}$$

STEP 2: PRIMAL UPDATE

$$w^{t+1} \leftarrow w^t - \sum_{i \in S_t} \frac{\theta}{n \lambda p_i} A_i \left(\nabla \phi_i(A_i^\top w^t) + \alpha_i^t\right)$$

$$p_i = \mathbf{P}(i \in S_t)$$

This is just maintaining the relationship

4.3
Complexity

ESO Assumption (same as before!)

Parameters $v_1, \ldots, v_n$ satisfy:

$$\mathbf{E}\left\|\sum_{i\in S_t} A_i\alpha_i\right\|^2 \leq \sum_{i=1}^{n} p_i v_i \|\alpha_i\|^2$$

inequality must hold for all

$$\alpha_1, \ldots, \alpha_n \in \mathbb{R}^m$$

$$p_i = \mathbf{P}(i \in S_t)$$

Complexity

Theorem [Csiba & R '15]

A constant depending on $P, w^0, \alpha_i^0, w^*, \alpha_i^*$

$$t \geq \max_i \left(\frac{1}{p_i} + \frac{v_i}{p_i \lambda \gamma n} \right) \log \left(\frac{C}{\epsilon} \right)$$

$p_i = \mathbf{P}(i \in S_t)$

$$\mathbf{E}\left[P(w^t) - P(w^*)\right] \leq \epsilon$$

4.4
Experiments

Some More Efficient Primal Methods for ERM: SAG, SVRG and S2GD

SAG: Stochastic Average Gradient

N. Le Roux, M. Schmidt, and F. Bach. **A stochastic gradient method with an exponential convergence rate for finite training sets**. *NIPS*, 2012

SVRG: Stochastic Variance Reduced Gradient

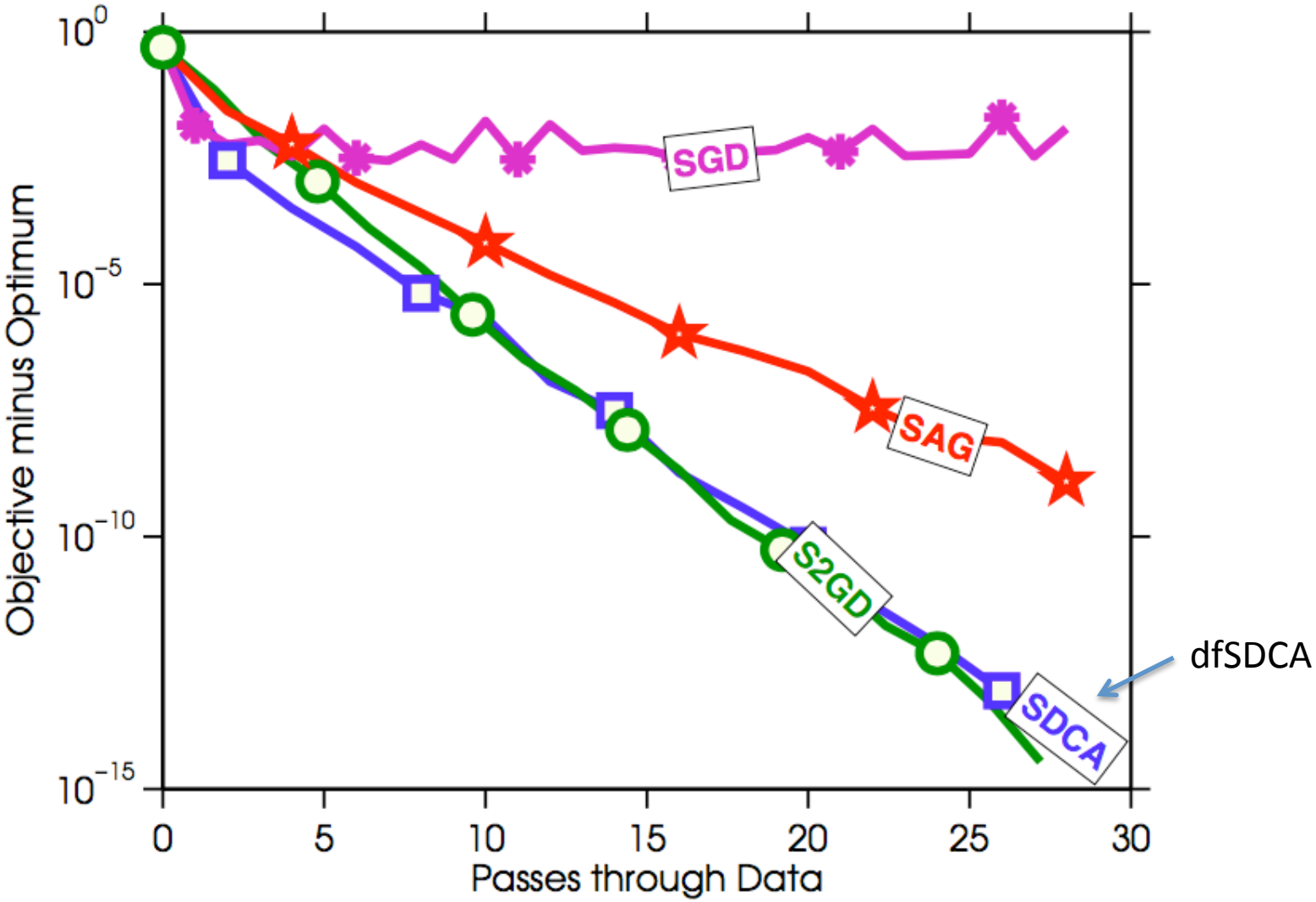
Rie Johnson and Tong Zhang. **Accelerating stochastic gradient descent using predictive variance reduction.** *NIPS*, 2013.

S2GD: Semi-Stochastic Gradient Descent

J. Konečný and P. R. **Semi-stochastic gradient descent methods.** *arXiv:1312.1666*, 2013

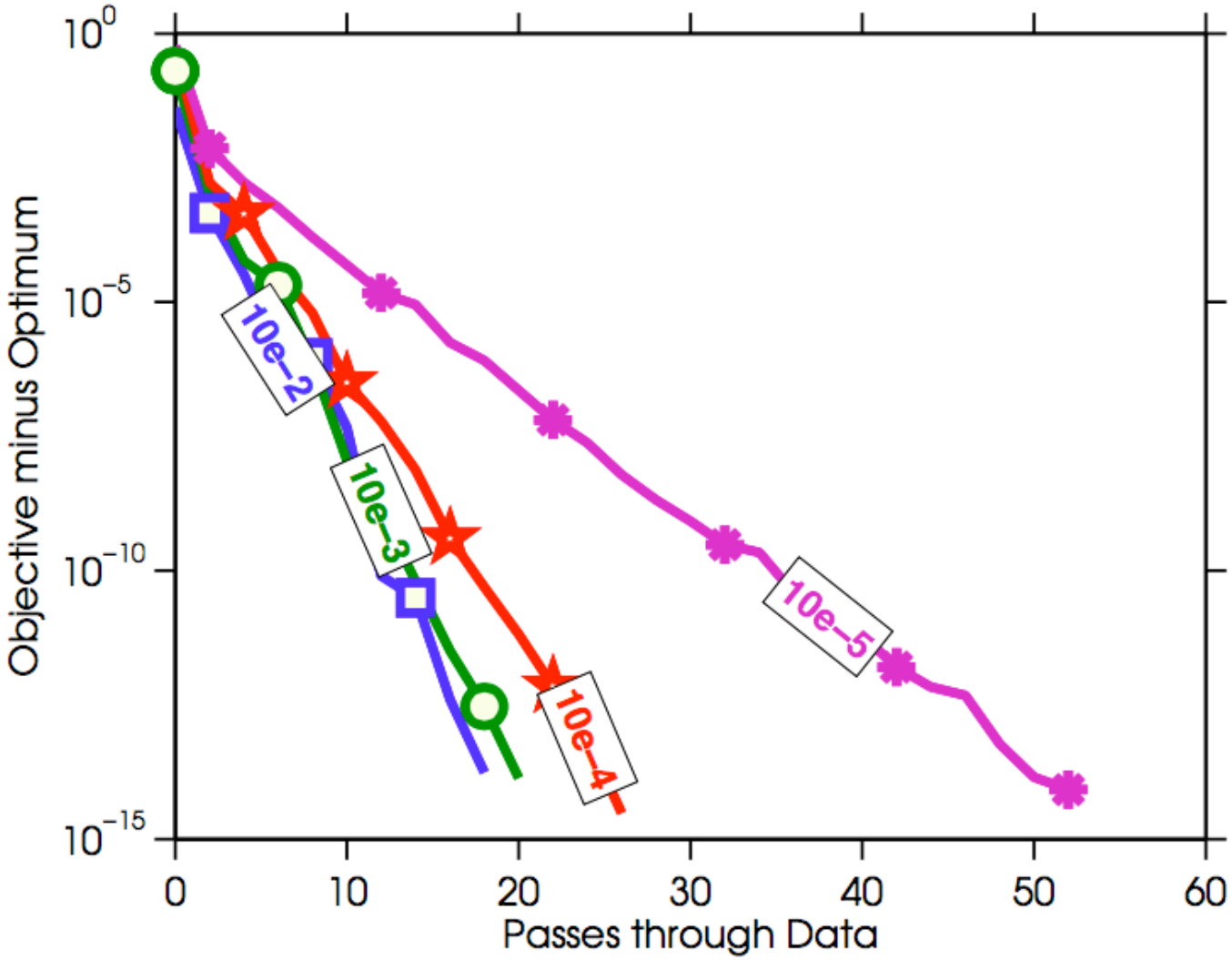
Modern Methods for ERM vs SGD

Dataset: rcv1 (n = 20,241 ; d = 47,232)



Behavior of dfSDCA for various λ

Dataset: rcv1 (n = 20,241 ; d = 47,232)



EXTRA MATERIAL
I WON'T HAVE TIME TO COVER

5. Distributed Optimization

References

P.R. and Martin Takáč. **Distributed coordinate descent for learning with big data.** *arXiv:1310.2059*, 2013

Olivier Fercoq, Zheng Qu, P.R. and Martin Takáč. **Fast distributed coordinate descent for minimizing non-strongly convex losses.** *In 2014 IEEE International Workshop on Machine Learning for Signal Processing*, 2014

Zheng Qu, P.R. and Tong Zhang. **Randomized dual coordinate ascent with arbitrary sampling.** *In NIPS 2015 (arXiv:1411.5873)*

5.1
Distributed Quartz

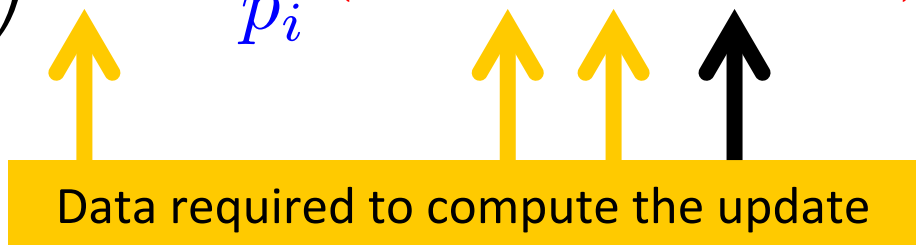
Distributed Quartz: Perform the Dual Updates in a Distributed Manner

Quartz STEP 2: DUAL UPDATE

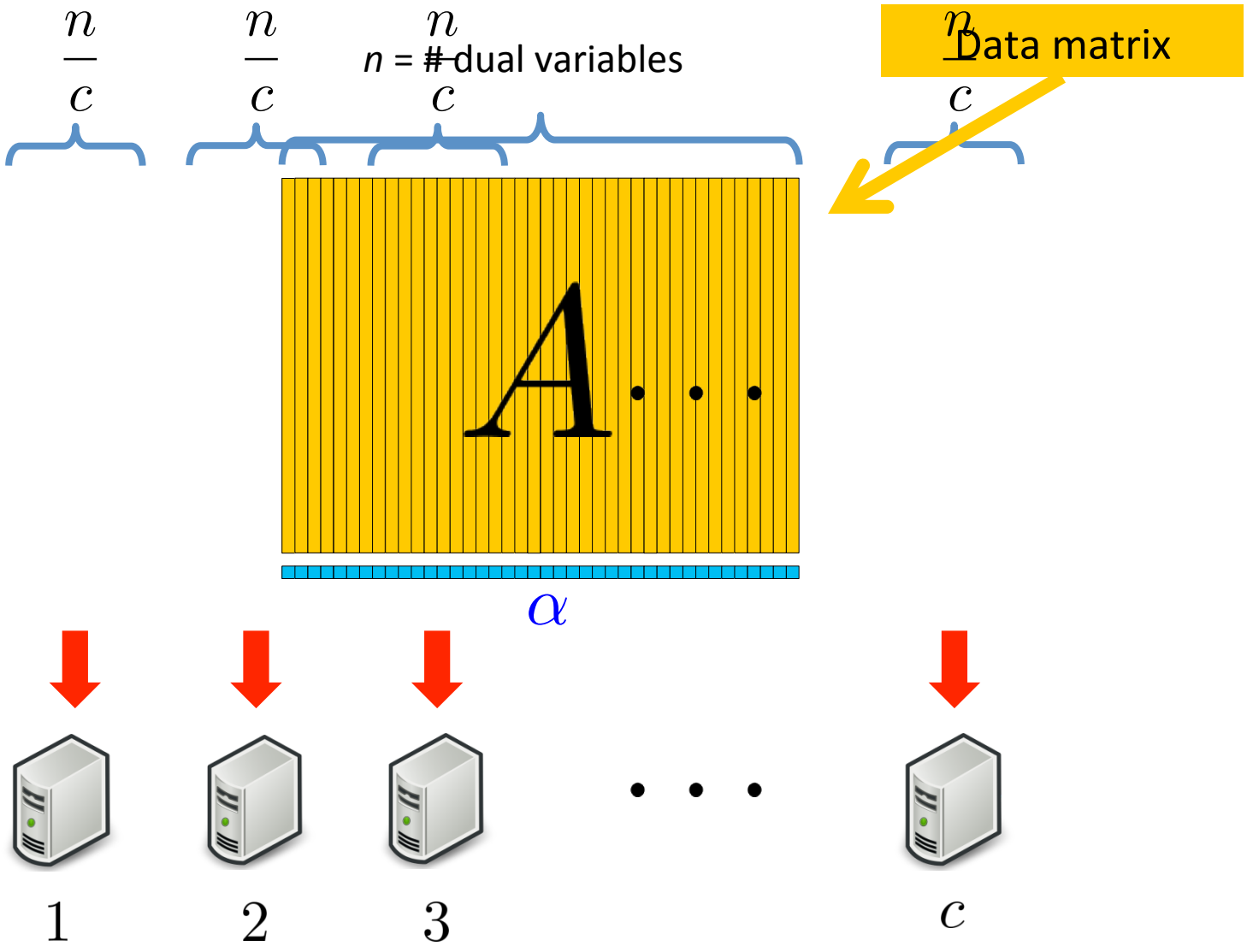
Choose a random set S_t of dual variables

For $i \in S_t$ do

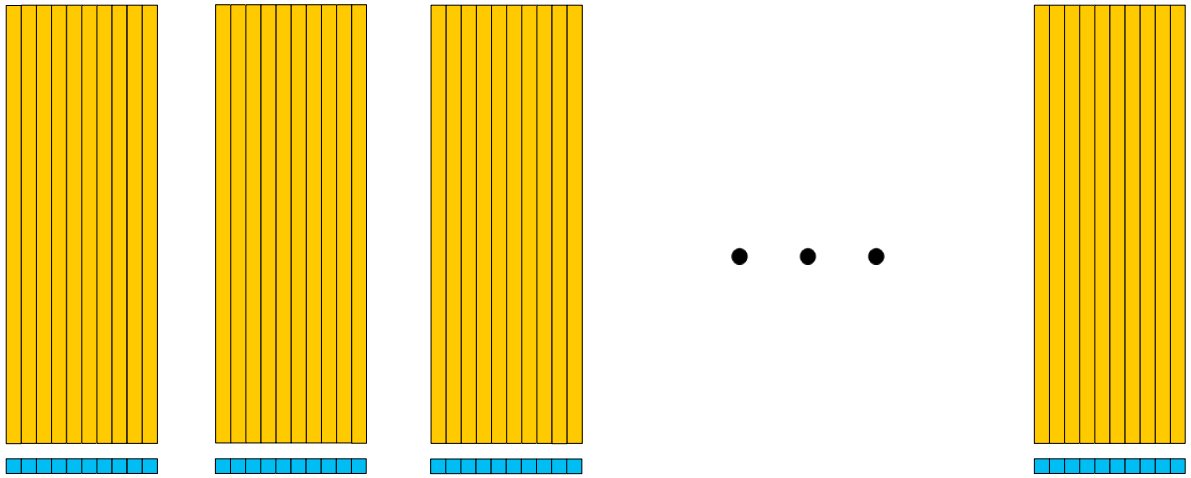
$$\alpha_i^{t+1} \leftarrow \left(1 - \frac{\theta}{p_i}\right)\alpha_i^t + \frac{\theta}{p_i}\left(-\nabla\phi_i(A_i^\top w^{t+1})\right)$$



Distribution of Data



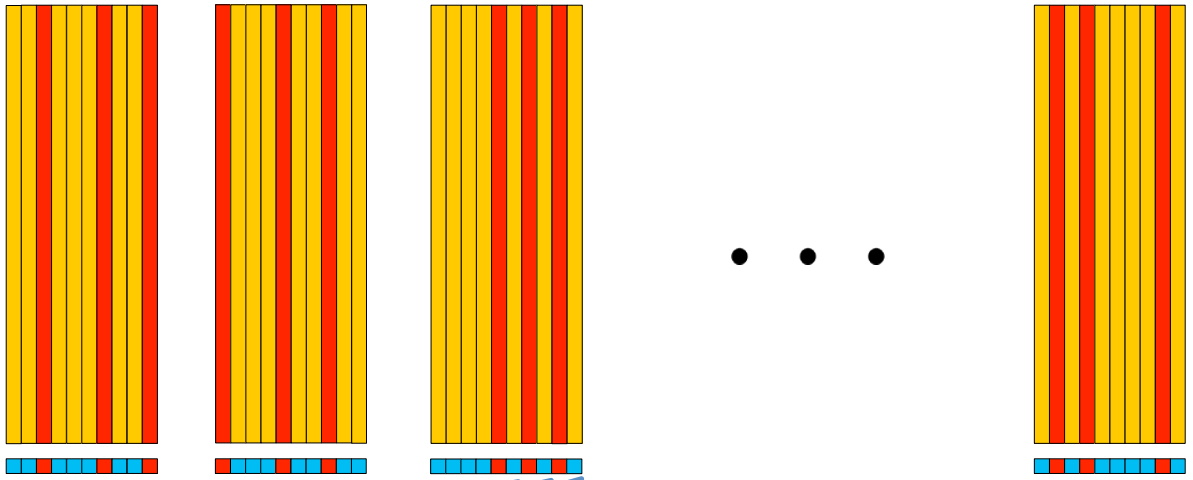
Distributed sampling



Distributed sampling

Each node independently picks τ dual variables from those it owns, uniformly at random

$\tau = 3$



$$\hat{S} = \{3, 7, 10\} \cup \{11, 15, 18\} \cup \{25, 27, 29\} \cup \cdots$$

Random set of dual variables

Also see: CoCoA+ [Ma, Smith, Jaggi et al 15]

5.2
Complexity

Complexity of Distributed Quartz

Key: Get the right stepsize parameters v (so that the ESO inequality holds)

The leading term in the complexity bound then is:

$$\max_i \left(\frac{1}{p_i} + \frac{v_i}{p_i \lambda \gamma n} \right)$$

$$||$$

$$\frac{n}{c\tau} + \frac{\text{Something that looks complicated}}{\lambda \gamma c \tau}$$

$$||$$

$$\frac{n}{c\tau} + \max_i \frac{\lambda_{\max}\left(\sum_{j=1}^{d} \left(1 + \frac{(\tau-1)(\omega_j - 1)}{\max\{n/c-1,1\}} + \left(\frac{\tau c}{n} - \frac{\tau - 1}{\max\{n/c-1,1\}} \right) \frac{\omega_j' - 1}{\omega_j'} \omega_j \right) A_{ji}^\top A_{ji} \right)}{\lambda \gamma c \tau}$$

5.3
Experiments

Experiment

Machine: 128 nodes of Hector Supercomputer (4096 cores)

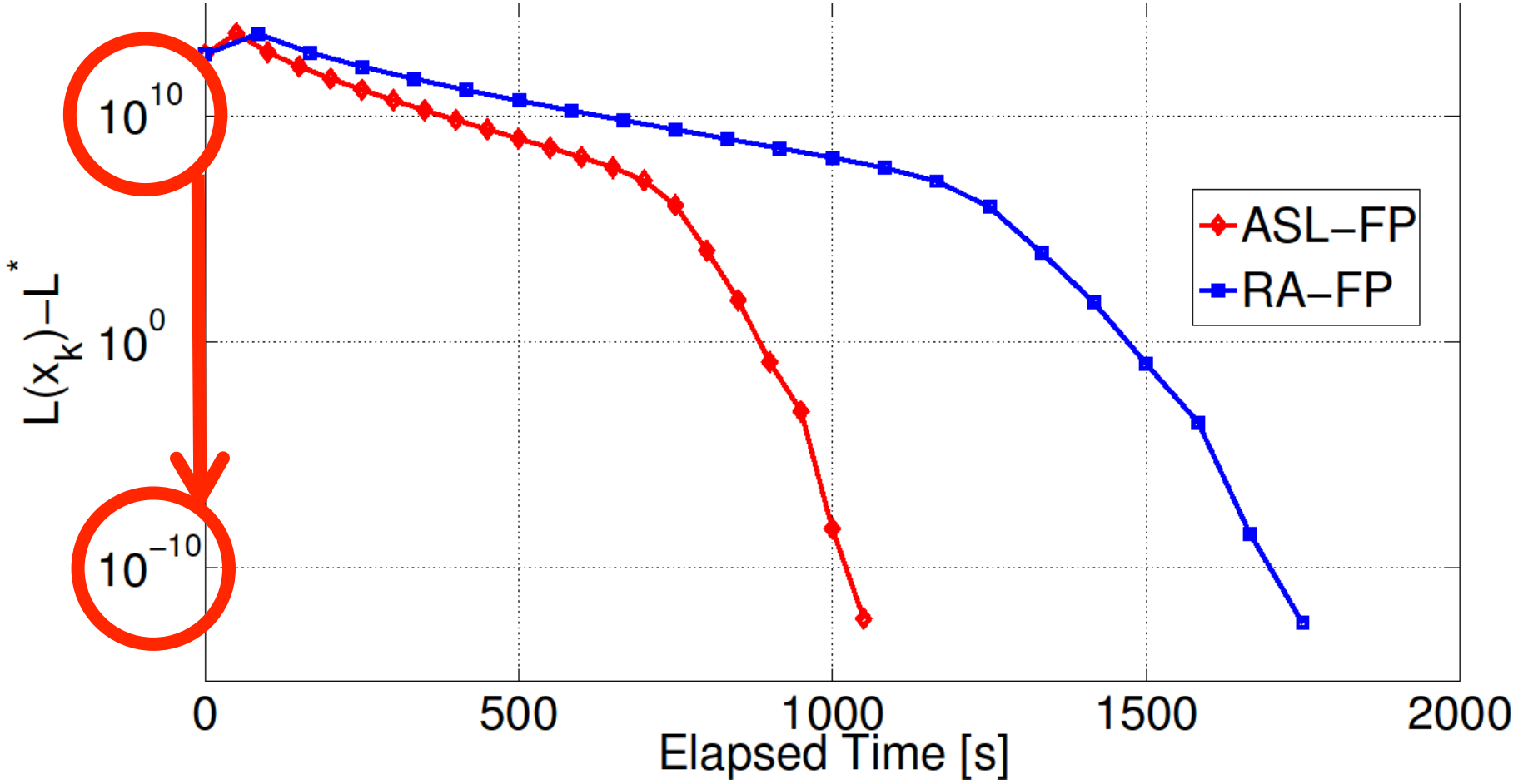
Problem: LASSO, n = 1 billion, d = 0.5 billion, 3 TB

Algorithm: 

with c = 512

P.R. and Martin Takáč. **Distributed coordinate descent for learning with big data.** *arXiv:1310.2059*, 2013

LASSO: 3TB data + 128 nodes



Experiment

Machine: 128 nodes of Archer Supercomputer

Problem: LASSO, n = 5 million, d = 50 billion, 5 TB
(60,000 nnz per row of A)

Algorithm 

with c = 256

Olivier Fercoq, Zheng Qu, P.R. and Martin Takáč. **Fast distributed coordinate descent for minimizing non-strongly convex losses.** *In 2014 IEEE International Workshop on Machine Learning for Signal Processing*, 2014

LASSO: 5TB data (d = 50 billion)
128 nodes